

Annals of Computer Science and Information Systems
Volume 30

Proceedings of the 17th Conference on Computer Science and Intelligence Systems

September 4–7, 2022. Sofia, Bulgaria



Maria Ganzha, Leszek Maciaszek, Marcin Paprzycki,
Dominik Ślęzak (eds.)

Annals of Computer Science and Information Systems, Volume 30

Series editors:

Maria Ganzha (Editor-in-Chief),

*Systems Research Institute Polish Academy of Sciences and Warsaw University of
Technology, Poland*

Leszek Maciaszek,

Wrocław University of Economy, Poland and Macquarie University, Australia

Marcin Paprzycki,

Systems Research Institute Polish Academy of Sciences and Management Academy, Poland

Dominik Ślęzak,

University of Warsaw, Poland

Senior Editorial Board:

Wil van der Aalst,

*Department of Mathematics & Computer Science, Technische Universiteit Eindhoven
(TU/e), Eindhoven, Netherlands*

Enrique Alba,

University of Málaga, Spain

Marco Aiello,

*Faculty of Mathematics and Natural Sciences, Distributed Systems, University of
Groningen, Groningen, Netherlands*

Mohammed Atiquzzaman,

School of Computer Science, University of Oklahoma, Norman, USA

Christian Blum,

Artificial Intelligence Research Institute (IIIA-CSIC), Barcelona, Spain

Jan Bosch,

Chalmers University of Technology, Gothenburg, Sweden

George Boustras,

European University, Cyprus

Barrett Bryant,

Department of Computer Science and Engineering, University of North Texas, Denton, USA

Rajkumar Buyya,

*Claud Computing and Distributed Systems (CLOUDS) Lab, University of Melbourne,
Australia*

Hristo Djidjev,

*Los Alamos National Laboratory, Los Alamos, NM, USA and Institute of Information and
Communication Technologies, Sofia, Bulgaria*

Włodzisław Duch,

*Department of Informatics, and NeuroCognitive Laboratory, Center for Modern
Interdisciplinary Technologies, Nicolaus Copernicus University, Toruń, Poland*

Hans-George Fill,

University of Fribourg, Switzerland

Ana Fred,

*Department of Electrical and Computer Engineering, Instituto Superior Técnico
(IST—Technical University of Lisbon), Lisbon, Portugal*

Janusz Górski,

Department of Software Engineering, Gdańsk University of Technology, Gdańsk, Poland

Giancarlo Guizzardi,

*Free University of Bolzano-Bozen, Italy, Senior Member of the Ontology and Conceptual
Modeling Research Group (NEMO), Brazil*

Francisco Herrera,

Dept. Computer Sciences and Artificial Intelligence Andalusian Research Institute in Data Science and Computational Intelligence (DaSCI) University of Granada, Spain

Mike Hinchey,

Lero—the Irish Software Engineering Research Centre, University of Limerick, Ireland

Janusz Kacprzyk,

Systems Research Institute, Polish Academy of Sciences, Warsaw, Poland

Irwin King,

The Chinese University of Hong Kong, Hong Kong

Juliusz L. Kulikowski,

Natęcz Institute of Biocybernetics and Biomedical Engineering, Polish Academy of Sciences, Warsaw, Poland

Michael Luck,

Department of Informatics, King's College London, London, United Kingdom

Jan Madey,

Faculty of Mathematics, Informatics and Mechanics at the University of Warsaw, Poland

Stan Matwin,

Dalhousie University, University of Ottawa, Canada and Institute of Computer Science, Polish Academy of Science, Poland

Marjan Mernik,

University of Maribor, Slovenia

Michael Segal,

Ben-Gurion University of the Negev, Israel

Andrzej Skowron,

Faculty of Mathematics, Informatics and Mechanics at the University of Warsaw, Poland

John F. Sowa,

VivoMind Research, LLC, USA

George Spanoudakis,

Research Centre for Adaptive Computing Systems (CeNACS), School of Mathematics, Computer Science and Engineering, City, University of London

Editorial Associates:

Katarzyna Wasielewska,

Systems Research Institute Polish Academy of Sciences, Poland

Paweł Sitek,

Kielce University of Technology, Kielce, Poland

TeXnical editor: Aleksander Denisiuk,

University of Warmia and Mazury in Olsztyn, Poland

Proceedings of the 17th Conference on Computer Science and Intelligence Systems

Maria Ganzha, Leszek Maciaszek, Marcin Paprzycki,
Dominik Ślęzak (eds.)



2022, Warszawa,
Polskie Towarzystwo
Informatyczne



2022, New York City,
Institute of Electrical and
Electronics Engineers

Annals of Computer Science and Information Systems, Volume 30

Proceedings of the 17th Conference on Computer Science and
Intelligence Systems

ART: ISBN 978-83-965897-1-2, IEEE Catalog Number CFP2285N-ART

USB: ISBN 978-83-965897-0-5, IEEE Catalog Number CFP2285N-USB

WEB: ISBN 978-83-962423-9-6

ISSN 2300-5963

DOI 10.15439/978-83-962423-9-6

© 2022, Polskie Towarzystwo Informatyczne

Ul. Solec 38/103

00-394 Warsaw, Poland

Contact: secretariat@fedcsis.org

<http://annals-csis.org/>

Cover art: Bażantarnia

Mariusz Edward Owczarek,

Elbląg, Poland

Also in this series:

Volume 32: Communication Papers of the 17th Conference on Computer Science and
Intelligence Systems, **ISBN WEB: 978-83-965897-4-3, ISBN USB: 978-83-965897-5-0**

Volume 31: Position Papers of the 17th Conference on Computer Science and
Intelligence Systems, **ISBN WEB: 978-83-965897-2-9, ISBN USB: 978-83-965897-3-6**

Volume 29: Recent Advances in Business Analytics. Selected papers of the 2021
KNOWCON-NSAIS workshop on Business Analytics **ISBN WEB: 978-83-962423-7-2,**
ISBN USB: 978-83-962423-6-5

Volume 28: Proceedings of the 2021 International Conference on Research in
Management & Technovation, **ISBN WEB: 978-83-962423-4-1, ISBN USB: 978-83-962423-5-8**

Volume 27: Proceedings of the Sixth International Conference on Research in Intelligent
and Computing in Engineering, **ISBN WEB: 978-83-962423-2-7, ISBN USB: 978-83-962423-3-4**

Volume 26: Position and Communication Papers of the 16th Conference on Computer
Science and Intelligence Systems, **ISBN WEB: 978-83-959183-9-1, ISBN USB: 978-83-962423-0-3**

Volume 25: Proceedings of the 16th Conference on Computer Science and Intelligence
Systems, **ISBN Web 978-83-959183-6-0, ISBN USB 978-83-959183-7-7, ISBN ART 978-83-959183-8-4**

Volume 24: Proceedings of the International Conference on Research in Management &
Technovation 2020, **ISBN WEB: 978-83-959183-5-3, ISBN USB: 978-83-959183-4-6**

Volume 23: Communication Papers of the 2020 Federated Conference on Computer
Science and Information Systems, **ISBN WEB: 978-83-959183-2-2, ISBN USB: 978-83-959183-3-9**

Volume 22: Position Papers of the 2020 Federated Conference on Computer Science and
Information Systems, **ISBN WEB: 978-83-959183-0-8, ISBN USB: 978-83-959183-1-5**

Volume 21: Proceedings of the 2020 Federated Conference on Computer Science and
Information Systems, **ISBN WEB 978-83-955416-7-4, ISBN USB 978-83-955416-8-1,**

ISBN ART 978-83-955416-9-8

Volume 20: Communication Papers of the 2019 Federated Conference on Computer
Science and Information Systems, **ISBN WEB: 978-83-955416-3-6, ISBN USB: 978-83-955416-4-3**

Volume 19: Position Papers of the 2019 Federated Conference on Computer Science and
Information Systems, **ISBN WEB: 978-83-955416-1-2, ISBN USB: 978-83-955416-2-9**

DEAR Reader, it is our pleasure to present to you Proceedings of the 17th Conference on Computer Science and Intelligence Systems (FedCSIS 2022), which took place on September 4-7, 2022, in Sofia, Bulgaria, and in the hybrid mode.

The main Conference Chair of FedCSIS 2022 was Stefka Fidanova, while Nina Dobrinkova acted as the Chair of the Organizing Committee. This year, FedCSIS was organized by the Polish Information Processing Society (Mazovia Chapter), IEEE Poland Section Computer Society Chapter, Systems Research Institute Polish Academy of Sciences, Warsaw University of Technology, Wrocław University of Economics and Institute of Information and Communication Technologies, Bulgarian Academy of Sciences.

FedCSIS 2022 was technically co-sponsored by: IEEE Bulgarian Section, IEEE Poland Section, IEEE Czechoslovakia Section Computer Society Chapter, IEEE Poland Section Systems, Man, and Cybernetics Society Chapter, IEEE Poland Section Computational Intelligence Society Chapter, IEEE Poland Section Control System Society Chapter, Committee of Computer Science of the Polish Academy of Sciences, Mazovia Cluster ICT, Poland and Bulgarian Section of SIAM.

Moreover, last year the FedCSIS conference series formed the strategic alliance with QED Software, a Polish software company developing AI-based technologies and acting as the technological co-founder in the AI-driven start-ups. This collaboration has been continued.

During FedCSIS 2022, the keynote lectures were delivered by:

- Krassimir Atanassov, Bulgarian Academy of Sciences, Sofia, Bulgaria: “*Remarks on Index Matrices*”,
- Chris Cornelis, Ghent University, Department of Applied Mathematics, Computer Science and Statistics: “*Hybridization of Fuzzy Sets and Rough Sets: Achievements and Opportunities*”,
- Ivan Lukovic, University of Belgrade, Faculty of Organizational Sciences, “*Organizational Capability for Information Management – Do We Feel a Big Data Crisis?*”,
- Stefano Mariani on behalf of Franco Zambonelli (due to serious health issues encountered right before the conference), University of Modena e Reggio Emilia, Italy: “*Individual and Collective Self-development*”.

Moreover, this year, two special guests delivered invited talks:

- Bogusław Cyganek, who was awarded the 2021 Wiley-IEEE Press Award, for his recent book “*Introduction to Programming with C++ for Engineers*”,
- Andrzej Skowron, who delivered the talk: “*Rough Sets Turn 40: From Information Systems to Intelligent Systems*”, which was devoted to the 40th anniversary of introduction, by late Professor Zdzisław Pawlak, of the theory of Rough Sets.

FedCSIS 2022 consisted of five Tracks and a special event for Doctoral School Students. Within each Track, topical Technical Sessions have been organized. Each Technical Session was split into fully on site and fully on-

line sub-sessions. The on-site part of the conference took place in the facilities of the Crisis Management and Disaster Response Centre of Excellence in Sofia, Bulgaria.

Some of Technical Sessions have been associated with the FedCSIS conference series for many years, while some of them are relatively new. The role of the technical Sessions is to focus and enrich discussions on selected areas, pertinent to the general scope of each Track. The list of Tracks, and topical Technical Sessions organized within their scope, was as follows.

- **Track 1: Advanced Artificial Intelligence in Applications (17th Symposium AAIA'22)**
 - Artificial Intelligence for Next-Generation Diagnostic Imaging (1st Workshop AI4NextGenDI'22)
 - Artificial Intelligence in Machine Vision and Graphics (4th Workshop AIMaViG'22)
 - Personalization and Recommender Systems (1st Workshop PeRS'22)
 - Rough Sets: Theory and Applications (4th International Symposium RSTA'22)
 - Computational Optimization (15th International Workshop WCO'22)
- **Track 2: Computer Science & Systems (CSS'22)**
 - Computer Aspects of Numerical Algorithms (15th Workshop CANA'22)
 - Concurrency, Specification and Programming (30th International Symposium CS&P'22)
 - Multimedia Applications and Processing (15th International Symposium MMAP'22)
 - Scalable Computing (12th Workshop WSC'22)
- **Track 3: Network Systems and Applications (NSA'22)**
 - Complex Networks - Theory and Application (1st Workshop CN-TA'22)
 - Internet of Things – Enablers, Challenges and Applications (6th Workshop IoT-ECAW'22)
 - Cyber Security, Privacy, and Trust (3rd International Forum NEMESIS'22)
- **Track 4: Advances in Information Systems and Technology (AIST'22)**
 - Data Science in Health, Ecology and Commerce (4th Workshop DSH'22)
 - Information Systems Management (17th Conference ISM'22)
 - Knowledge Acquisition and Management (28th Conference KAM'22)
- **Track 5: Software and System Engineering (S3E'22)**
 - Cyber-Physical Systems (9th International Workshop IWCPS'22)
 - Model Driven Approaches in System Development (7th Workshop MDASD'22)
 - Software Engineering (42nd IEEE Workshop SEW'22)
- **7th Doctoral Symposium on Recent Advances in Information Technology (DS-RAIT'22)**

Each contribution, found in this volume, was refereed by at least two referees and the acceptance rate of regular full papers was slightly below 19% (55 accepted contributions, out of 290 general submissions).

During FedCSIS 2022, for the second time, the Zdzisław Pawlak award was presented to the best papers from across the whole conference. It was done to further underscore the fact that scientific achievements of Professor Pawlak had gone far beyond Artificial Intelligence. The following contributions have been awarded in three categories:

- In the category **Best Paper**: Kamil Rybiński (Warsaw University of Technology, Poland) and Michał Śmiąlek (Warsaw University of Technology, Poland) for the paper: “*Beyond Low-Code Development: Marrying Requirements Models and Knowledge Representations*”.
- In the category **Industry Cooperation**: Ghada Moualla (Orange Labs, France), Sebastien Bolle (Orange Labs, France), Marc Douet (Orange Labs, France) and Eric Rutten (INRIA, France) for the paper: “*Self-adaptive Device Management for the IoT Using Constraint Solving*”.
- In the category **Young Researcher**: Eyad Kannout (University of Warsaw, Poland), Michał Grodzki (University of Warsaw, Poland) and Marek Grzegorowski (University of Warsaw, Poland) for the paper: “*Utilizing Frequent Pattern Mining for Solving Cold-Start Problem in Recommender Systems*”.

Moreover, the Award Committee decided to fund two extra **Young Researcher** distinctions:

- Aleksandra Bączkiewicz (University of Szczecin, Poland) for the paper: “*Towards Temporal Multi-Criteria Assessment of Sustainable RES Exploitation in European Countries*”.
- Aleksandra Weiss (Scientific Circle of Robotics UWM in Olsztyn, Poland), Marcin Młyński (Scientific Circle of Robotics UWM in Olsztyn, Poland) and Piotr Artiemjew (University of Warmia and Mazury, Poland) for the paper: “*About Classifiers Quality Assessment: Balanced Accuracy Curve (BAC) as an alternative for ROC and PR Curve*”.

All the above awards and distinctions were jointly sponsored by the Mazovia Branch of the Polish Information Processing Society and by QED Software.

FedCSIS’22 also hosted the **Data Mining Competition** (already for the 5th time during the history of the FedCSIS conferences) co-organized by QED Software at the online platform KnowledgePit, and sponsored by Control System Software, a Polish company which develops decision support and optimization systems in the areas of transportation, spedition, and logistics.

The competition attracted 130 teams from 24 countries. The papers describing the most interesting solutions submitted to the competition were included in the FedCSIS 2022 proceedings in the special section of **Track 1**. The following papers represent the best solutions awarded by Control System Software:

- Paper describing **the winning solution**: Dymitr Ruta (Khalifa University, UAE), Ming Liu (Khalifa University,

UAE), Ling Cen (Khalifa University, UAE), Quang Hieu Vu (VNG Corporation, Vietnam), “*Diversified gradient boosting ensembles for prediction of the cost of forwarding contracts*”

- Paper describing **the 2nd best solution**: Haitao Xiao, (Chinese Academy of Sciences, China), Yuling Liu (Chinese Academy of Sciences, China), Dan Du (Chinese Academy of Sciences, China), Zhigang Lu (Chinese Academy of Sciences, China), “*An Approach for Predicting the Costs of Forwarding Contracts using Gradient Boosting*”
- Paper describing **the 3rd best solution**: Quang Hieu Vu (VNG Corporation, Vietnam), Ling Cen (Khalifa University, UAE), Dymitr Ruta (Khalifa University, UAE), Ming Liu (Khalifa University, UAE), “*Key Factors to Consider when Predicting the Costs of Forwarding Contracts*”
- Paper describing **the most practical solution**: Sławomir Pioroński (Adam Mickiewicz University, Poland), Tomasz Górecki (Adam Mickiewicz University, Poland), “*Using gradient boosting trees to predict the costs of forwarding contracts*”

The program of FedCSIS 2022 required a dedicated effort of many people. We would like to express our warmest gratitude to all Committee members, of each Track and each Technical Session, for their hard work in attracting and later refereeing 290 submissions.

We thank the authors of papers for their great contribution to the theory and practice of computing and intelligence systems. We are grateful to the invited speakers for sharing their knowledge and wisdom with the participants.

Last, but not least, we thank Stefka Fidanova and Nina Dobrinkova. It should be stressed that they made all the preparations to organize the conference in Bulgaria for three years in a row, while only in 2022 the conference actually happened there. They also worked with us diligently to adapt the conference formula to organize it in hybrid mode. We are very grateful for all your efforts!

We hope that you had an inspiring conference. We also hope to meet you again for the 18th Conference on Computer Science and Intelligence Systems (FedCSIS 2023) which will take place in Warsaw, Poland on September 17-20, 2023.

Co-Chairs of the FedCSIS Conference Series:

Maria Ganzha, Warsaw University of Technology, Poland, and Systems Research Institute Polish Academy of Sciences, Warsaw, Poland

Leszek Maciaszek, Macquarie University, Sydney, Australia and Wrocław University of Economics, Wrocław, Poland

Marcin Paprzycki, Systems Research Institute Polish Academy of Sciences, Warsaw, Poland, and Warsaw Management University, Warsaw, Poland

Dominik Ślęzak, Institute of Informatics, University of Warsaw, Poland, and QED Software, Warsaw, Poland

Proceedings of the 17th Conference on Computer Science and Intelligence Systems

September 4–7, 2022. Sofia, Bulgaria

TABLE OF CONTENTS

CONFERENCE INVITED CONTRIBUTIONS

KEYNOTE PAPERS

- Two new operations over extended index matrices and their applications in Big Data** 1
Krassimir Atanassov, Veselina Bureva
- Hybridization of Fuzzy Sets and Rough Sets: Achievements and Opportunities** 7
Chris Cornelis
- Individual and Collective Self-Development: Concepts and Challenges** 15
Marco Lippi, Stefano Mariani, Matteo Martinelli, Franco Zambonelli

ANNIVERSARY PAPER

- Rough Sets Turn 40: From Information Systems to Intelligent Systems** 23
Andrzej Skowron, Dominik Ślęzak

IEEE AWARD PAPER

- Modern C++ in the era of new technologies and challenges—why and how to teach modern C++?** 35
Bogusław Cyganek

17TH INTERNATIONAL SYMPOSIUM ON ADVANCED ARTIFICIAL INTELLIGENCE IN APPLICATIONS

- Call For Papers** 41
- A novel link prediction approach on clinical knowledge graphs utilizing graph structures** 43
Jens Dörpinghaus, Tobias Hübenthal, Jennifer Faber
- Deep Learning Transformer Architecture for Named Entity Recognition on Low Resourced Languages: State of the art results** 53
Ridewaan Hanslo
- Demand forecasting in the fashion business — an example of customized nearest neighbour and linear mixed model approaches** 61
Joanna Henzel, Łukasz Wawrowski, Anna Kubina, Marek Sikora, Łukasz Wróbel
- Improving Re-rankCCP with Rules Quality Measures** 67
Piotr Jezusek, Aleksandra Karpus
- Using Transformer models for gender attribution in Polish** 73
Karol Kaczmarek, Jakub Pokrywka, Filip Graliński
- A Comparative Study of Short Text Classification with Spiking Neural Networks** 79
Piotr S. Maciąg, Wojciech Sitek, Łukasz Skonieczny, Henryk Rybiński

Rule-based approximation of black-box classifiers for tabular data to generate global and local explanations	89
<i>Cezary Maszczyk, Michal Kozielski, Marek Sikora</i>	
Automatic detection of potential customers by opinion mining and intelligent agents	93
<i>Raúl Moreno, Alberto Fernandez-Isabel, Isaac Martín De Diego, Javier M. Moguerza, Carmen Lancho, Marina Cuesta Santa Teresa</i>	
An Automated Algorithm for Fruit Image Dataset Building	103
<i>Horea-Bogdan Mureşan</i>	
NiaNet: A framework for constructing Autoencoder architectures using nature-inspired algorithms	109
<i>Sašo Pavlič, Iztok Fister Jr., Sašo Karakatič</i>	
Aspects of autonomous drive control using NVIDIA Jetson Nano microcomputer	117
<i>Kacper Podbucki, Tomasz Marciniak</i>	
Temporal Language Modeling for Short Text Document Classification with Transformers	121
<i>Jakub Pokrywka, Filip Graliński</i>	
An End-to-end Machine Learning System for Mitigating Checkout Abandonment in E-Commerce	129
<i>Md Rifatul Islam Rifat, Md Nur Amin, Mahmud Hasan Munna, Abdullah Al Imran</i>	
Reinforcement Learning for on-line Sequence Transformation	133
<i>Grzegorz Rypeś, Łukasz Lepak, Paweł Wawrzyński</i>	
Applying SoftTriple Loss for Supervised Language Model Fine Tuning	141
<i>Witold Sosnowski, Anna Wróblewska, Piotr Gawrysiak</i>	
About Classifiers Quality Assessment: Balanced Accuracy Curve (BAC) as an alternative for ROC and PR Curve	149
<i>Aleksandra Weiss, Marcin Młyński, Piotr Artiemjew</i>	
Extending Word2Vec with Domain-Specific Labels	157
<i>Miloš Švaňa</i>	

1ST WORKSHOP ON ARTIFICIAL INTELLIGENCE FOR NEXT-GENERATION DIAGNOSTIC IMAGING

Call For Papers	161
Development of an AI-based audiogram classification method for patient referral	163
<i>Michał Kassjański, Marcin Kulawiak, Tomasz Przewoźny</i>	
Canine age classification using Deep Learning as a step towards preventive medicine in animals	169
<i>Szymon Mazurek, Maciej Wielgosz, Jakub Caputa, Rafał Frączek, Michał Karwatowski, Jakub Grzeszczyk, Daria Łukasik, Anna Śmiech, Paweł Russek, Ernest Jamro, Agnieszka Dąbrowska-Boruch, Marcin Pietroń, Sebastian Koryciak, Kazimierz Wiatr</i>	

4TH INTERNATIONAL WORKSHOP ON ARTIFICIAL INTELLIGENCE IN MACHINE VISION AND GRAPHICS

Call For Papers	173
On the Feasible Regions Delimiting Natural Human Postures in a Novel Skeletal Representation	175
<i>Simon Hengeveld, Antonio Mucherino</i>	
A lightweight approach to two-person interaction classification in sparse image sequences	181
<i>Włodzimirz Kasprzak, Van Khanh Do, Paweł Piwowarski</i>	
Geometry-Aware Keypoint Network: Accurate Prediction of Point Features in Challenging Scenario	191
<i>Tomasz Nowak, Piotr Skrzypczyński</i>	
Analysis of Brain Tumor Using MRI Images	201
<i>Lewi Uberg, Seifedine Kadry</i>	

Insights into Neural Architectures for Learning Numerical Concepts from Simple Visual Data	205
<i>Andrzej Śluzek</i>	

1ST WORKSHOP ON PERSONALIZATION AND RECOMMENDER SYSTEMS

Call For Papers	211
User Experience and Multimodal Usability for Navigation Systems	213
<i>Lumbardha Hasimi, Aneta Poniszewska-Marańda</i>	
Utilizing Frequent Pattern Mining for Solving Cold-Start Problem in Recommender Systems	217
<i>Eyad Kannout, Michał Grodzki, Marek Grzegorowski</i>	
Learning edge importance in bipartite graph-based recommendations	227
<i>Robert Kwieciński, Tomasz Górecki, Agata Filipowska</i>	
Personality Prediction from Social Media Posts using Text Embedding and Statistical Features	235
<i>Seiyu Majima, Konstantin Markov</i>	

4TH INTERNATIONAL SYMPOSIUM ON ROUGH SETS: THEORY AND APPLICATIONS

Call For Papers	241
Three-way Learnability: A Learning Theoretic Perspective on Three-way Decision	243
<i>Andrea Campagner, Davide Ciucci</i>	
On Multiplicative, Additive and Qualitative Pairwise Comparisons	247
<i>Ryszard Janicki, Mahmoud Mahmoud</i>	
Malware Evolution and Detection Based on the Variable Precision Rough Set Model	253
<i>Manel Jerbi, Zaineb Chelly Dagdia, Slim Bechikh, Lamjed Ben Said</i>	
Multi-Criteria Decision-Making with Linguistic Labels	263
<i>Alicja Mieszkowicz-Rolka, Leszek Rolka</i>	
Fuzzy Quantifier-Based Fuzzy Rough Sets	269
<i>Adnan Theerens, Chris Cornelis</i>	
Feature Selection and Ranking Method based on Intuitionistic Fuzzy Matrix and Rough Sets	279
<i>Bich Khue Vo, Hung Son Nguyen</i>	

15TH INTERNATIONAL WORKSHOP ON COMPUTATIONAL OPTIMIZATION

Call For Papers	289
GaMeDE2 — improved Gap-based Memetic Differential Evolution applied to multi-modal optimisation	291
<i>Michał Antkiewicz, Paweł Myszkowski, Maciej Laszczyk</i>	
A chance-constraint approach for optimizing social engagement-based services	301
<i>Michel Bierlaire, Edoardo Fadda, Lohic Fotio Tiotsop, Daniele Manerba</i>	
Independent Component Analysis Based on Jacobi Iterative Framework and L1-norm Criterion	305
<i>Adam Borowicz</i>	
The electric vehicle shortest path problem with time windows and prize collection	313
<i>Antonio Cassia, Ola Jabali, Federico Malucelli, Marta Pascoal</i>	
Analyzing longitudinal Data in Knowledge Graphs utilizing shrinking pseudo-triangles	323
<i>Jens Dörpinghaus, Vera Weil, Johanna Binnewitt</i>	
Agricultural System Modelling with Ant Colony Optimization	329
<i>Stefka Fidanova, Ivan Dimov, Denitsa Angelova, Maria Ganzha</i>	

A GPU approach to distance geometry in 1D: an implementation in C/CUDA	333
<i>Simon B. Hengeveld, Antonio Mucherino</i>	
A Multi-objective Cluster-based Biased Random-Key Genetic Algorithm with Online Parameter Control Applied to Protein Structure Prediction	337
<i>Felipe Marchi, Rafael Stubs Parpinelli</i>	
Team Orienteering Problem with Time Windows and Variable Profit	347
<i>Eliseo Marzal, Laura Sebastia</i>	
Subcaterpillar Isomorphism: Subtree Isomorphism Restricted Pattern Trees To Caterpillars	351
<i>Tomoya Miyazaki, Kouichi Hirata</i>	
Improving N-NEH+ algorithm by using Starting Point method	357
<i>Radostaw Puka, Bartosz Łamasz, Iwona Skalna</i>	
Boosting a Genetic Algorithm with Graph Neural Networks for Multi-Hop Influence Maximization in Social Networks	363
<i>Camilo Chacón Sartori, Christian Blum</i>	
Stackelberg Strategies for Weighted Load Balancing Games	373
<i>Neta Stein, Tami Tamir</i>	
Intuitionistic Fuzzy Model of the Hungarian Algorithm for the Salesman Problem and Software Analysis of a Shipping Company Example	383
<i>Velichka Traneva, Deyan Mavrov, Stoyan Tranev</i>	
Software Implementation of the Optimal Temporal Intuitionistic Fuzzy Algorithm for Franchisee Selection	387
<i>Velichka Traneva, Deyan Mavrov, Stoyan Tranev</i>	
<hr/>	
DATA MINING COMPETITON	
Call For Papers	391
KnowledgePit Meets BrightBox: A Step Toward Insightful Investigation of the Results of Data Science Competitions	393
<i>Andrzej Janusz, Dominik Ślęzak</i>	
Predicting the Costs of Forwarding Contracts: Analysis of Data Mining Competition Results	399
<i>Andrzej Janusz, Antoni Jamiołkowski, Michał Okulewicz</i>	
Considering various aspects of models' quality in the ML pipeline - application in the logistics sector	403
<i>Eyad Kannout, Michał Grodzki, Marek Grzegorowski</i>	
Prediction of the Costs of Forwarding Contracts with Machine Learning Methods	413
<i>Stanisław Kaźmierczak</i>	
XGBoost meets TabNet in Predicting the Costs of Forwarding Contracts	417
<i>Aleksandra Lewandowska</i>	
Using gradient boosting trees to predict the costs of forwarding contracts	421
<i>Sławomir Pioroński, Tomasz Górecki</i>	
Predicting the Costs of Forwarding Contracts Using XGBoost and a Deep Neural Network	425
<i>Lukasz Podlódowski, Marek Kozłowski</i>	
Diversified gradient boosting ensembles for prediction of the cost of forwarding contracts	431
<i>Dymitr Ruta, Ming Liu, Ling Cen, Quang Hieu Vu</i>	
Application of Diversified Ensemble Learning in Real-life Business Problems: The Case of Predicting Costs of Forwarding Contracts	437
<i>Milena Trajanoska, Pavel Gjorgovski, Eftim Zdravevski</i>	
Key Factors to Consider when Predicting the Costs of Forwarding Contracts	447
<i>Quang Hieu Vu, Ling Cen, Dymitr Ruta, Ming Liu</i>	
An Approach for Predicting the Costs of Forwarding Contracts using Gradient Boosting	451
<i>Haitao Xiao, Yuling Liu, Dan Du, Zhigang Lu</i>	

COMPUTER SCIENCE AND SYSTEMS

Call For Papers	455
Block Subspace Iteration Method for Structural Analysis on Multicore Computers <i>Sergiy Fialko</i>	457
Flexible user query order for the speculative query support in RDBMS <i>Anna Sasak-Okoń</i>	467

15TH WORKSHOP ON COMPUTER ASPECTS OF NUMERICAL ALGORITHMS

Call For Papers	473
Automatic code optimization for computing the McCaskill partition functions <i>Włodzimierz Bielecki, Marek Palkowski, Maciej Poliwoda</i>	475
Influence of loop transformations on performance and energy consumption of the multithreaded WZ factorization <i>Beata Bylina, Jarosław Bylina, Monika Piekarcz</i>	479
A short note on post-hoc testing using random forests algorithm: Principles, asymptotic time complexity analysis, and beyond <i>Lubomír Štěpánek, Filip Habarta, Ivana Mala, Luboš Marek</i>	489

30TH INTERNATIONAL SYMPOSIUM ON CONCURRENCY, SPECIFICATION AND PROGRAMMING

Call For Papers	499
An observation on pure strategies in Security Games <i>Marek Adrian, Gianluigi Greco</i>	501
Formal analysis of timeliness in the RaSTA protocol <i>Billy Naumann, Christine Jakobs, Matthias Werner</i>	505
Channel-Less Process Communication <i>Tomas Plachetka</i>	515
Improvement of design anti-pattern detection with spatio-temporal rules in the software development process <i>Łukasz Puławski</i>	521

15TH INTERNATIONAL SYMPOSIUM ON MULTIMEDIA APPLICATIONS AND PROCESSING

Call For Papers	529
A Modified ICP Algorithm Based on FAST and Optical Flow for 3D Registration <i>Konrad Koniarski, Andrzej Myśliński</i>	531
Hierarchical data structures in rendering scenes containing a massive number of light sources <i>Andrzej Lamecki, Krzysztof Kaczmarek, Joanna Porter-Sobieraj</i>	535
Encrypting JPEG-compressed images by substituting Huffman code words <i>Marek Parfieniuk</i>	545

12TH WORKSHOP ON SCALABLE COMPUTING

Call For Papers	551
Khaos: Dynamically Optimizing Checkpointing for Dependable Distributed Stream Processing <i>Morgan Geldenhuys, Ben Pfister, Dominik Scheinert, Lauritz Thamsen, Odej Kao</i>	553
Increasing data availability and fault tolerance for decentralized collaborative data-sharing systems <i>Kamil Jarosz, Łukasz Opióła, Łukasz Dutka, Renata G. Słota, Jacek Kitowski</i>	563

NETWORK SYSTEMS AND APPLICATIONS

Call For Papers	567
An Integer Programming Approach Reinforced by a Message-passing Procedure for Detecting Dense Attributed Subgraphs	569
<i>Arman Ferdowsi</i>	
Location Accuracy of a Ground Station based on RSS in the Rice Channel	577
<i>Jarostaw Michalak</i>	

1ST WORKSHOP ON COMPLEX NETWORKS: THEORY AND APPLICATION

Call For Papers	581
The effectiveness analysis of selected IT tools for predictions of the COVID-19 pandemic	583
<i>Paweł Dymora, Mirosław Mazurek, Kamil Łyczko</i>	
Denotational Model and Implementation of Scalable Virtual Machine in CPDev	587
<i>Jan Sadolewski, Bartosz Trybus</i>	
Room mapping system using RFID and mobile robots	593
<i>Mariusz Skoczylas, Łukasz Gotówko, Mateusz Salach, Bartosz Trybus, Marcin Hubacz, Bartosz Pawłowicz</i>	

6TH WORKSHOP ON INTERNET OF THINGS - ENABLERS, CHALLENGES AND APPLICATIONS

Call For Papers	599
Wireless Agent-based Distributed Sensor Tuple Spaces using Bluetooth and IP Broadcasting	601
<i>Stefan Bosse</i>	
Resource Partitioning in Phoenix-RTOS for Critical and Noncritical Software for UAV systems	605
<i>Hubert Buczyński, Paweł Pisarczyk, Krzysztof Cabaj</i>	
Data Exchange Protocol for Cryptographic Key Distribution System Using MQTT Service	611
<i>Janusz Furtak</i>	
Formal verification of security properties of the Lightweight Authentication and Key Exchange Protocol for Federated IoT devices	617
<i>Michał Jarosz, Konrad Wrona, Zbigniew Zieliński</i>	
Secure Onboarding and Key Management in Federated IoT Environments	627
<i>Krzysztof Kanciak, Konrad Wrona, Michał Jarosz</i>	
Anomaly detection on compressed data in resource-constrained smart water meters	635
<i>Sarah Klein, Anna Hristoskova, Annanda Rath, Renaud Gonc</i>	
Self-adaptive Device Management for the IoT Using Constraint Solving	641
<i>Ghada Moualla, Sebastien Bolle, Marc Douet, Eric Rutten</i>	

3RD INTERNATIONAL FORUM ON CYBER SECURITY, PRIVACY AND TRUST

Call For Papers	651
Heuristic Risk Treatment for ISO/SAE 21434 Development Projects	653
<i>Christine Jakobs, Matthias Werner, Karsten Schmidt, Gerhard Hansch</i>	
Universal Key to Authentication Authority with Human-Computable OTP Generator	663
<i>Sławomir Matelski</i>	
Low-complexity access control scheme for MEC-based services	673
<i>Mariusz Sepczuk, Zbigniew Kotulski, Wojciech Niewolski, Tomasz Nowak</i>	

ADVANCES IN INFORMATION SYSTEMS AND TECHNOLOGIES

Call For Papers	683
A Blockchain-Based Self-Sovereign Identity Approach for Inter-Organizational Business Processes	685
<i>Amal Abid, Saoussen Cheikhrouhou, Slim Kallel, Mohamed Jmaiel</i>	
Students' Online Behaviour in the Time of the COVID-19 Pandemic: Insights from Poland and Ukraine	695
<i>Dariusz Dymek, Svitlana Didkivska, Mariusz Grabowski, Grażyna Paliwoda-Pękosz, Tetiana Anatoliivna Vakaliuk</i>	
A Look at Evolution of Teams, Society, Smart Cities, and Information Systems based on Patterns of Primary, Adaptable, Information, and Creative Society	701
<i>Dmitriy Gakh</i>	
Identifying Reliable Sources of Information about Companies in Multilingual Wikipedia	705
<i>Włodzimierz Lewoniewski, Krzysztof Węcel, Witold Abramowicz</i>	
Modelling an IT solution to anonymise selected data processed in digital documents	715
<i>Barbara Probierz, Tomasz Jach, Jan Kozak, Radosław Pacud, Tomasz Turek</i>	
Performance Management of IT Professionals: A Humanistic Model	721
<i>Marcus Vinicius Alencar Terra, Vanessa Tavares de Oliveira Barros, Rodolfo Miranda de Barros</i>	

4TH WORKSHOP ON DATA SCIENCE IN HEALTH, ECOLOGY AND COMMERCE

Call For Papers	731
Encoder-Decoder Neural Network with Attention Mechanism for Types Detection in Linked Data	733
<i>Oussama Hamel, Messaouda Fareh</i>	
Multi-agent model of trust dissemination based on optimistic and pessimistic fuzzy aggregation norms	741
<i>Aleksandra Mrela, Oleksandr Sokolov, Maryla Bieniek-Majka, Veslava Osinksa, Włodzisław Duch</i>	
Detecting Symptoms of Dementia in Elderly Persons using Features of Pupil Light Reflex	745
<i>Minoru Nakayama, Wioletta Nowak, Anna Zarowska</i>	
Artificial Intelligence in Personalized Healthcare Analysis for Womens' Menstrual Health Disorders	751
<i>Łukasz Sosnowski, Joanna Żuławińska, Soma Dutta, Iwona Szymusik, Aleksandra Zyguła, Elżbieta Bambul-Mazurek</i>	

17TH CONFERENCE ON INFORMATION SYSTEMS MANAGEMENT

Call For Papers	761
Optimization of Processes for Shared Cars	763
<i>Janis Bicevskis, Ivo Odītis, Zane Bicevska, Viesturs Spulis</i>	
Towards Temporal Multi-Criteria Assessment of Sustainable RES Exploitation in European Countries	769
<i>Aleksandra Bączkiewicz</i>	
Process-oriented documentation of user requirements for analytical applications - challenges, state of the art and evaluation of a service-based configuration approach	773
<i>Christian Hrach, Rainer Alt, Stefan Sackmann</i>	
A novel iterative approach to determining compromise rankings	783
<i>Bartłomiej Kizielewicz, Andrii Shekhovtsov, Wojciech Sałabun</i>	
Towards the identification of MARCOS models based on intuitionistic fuzzy score functions	789
<i>Bartłomiej Kizielewicz, Bartosz Paradowski, Jakub Więckowski, Wojciech Sałabun</i>	

Towards Sustainable Transport Assessment Considering Alternative Fuels Based on MCDA Methods	799
<i>Jarosław Wątróbski, Aleksandra Bączkiewicz</i>	
Critical Success Factors for Adopting Electronic Document Management Systems in Government Units	809
<i>Ewa Ziemia, Tomasz Papaj, Danuta Descours</i>	

28TH CONFERENCE ON KNOWLEDGE ACQUISITION AND MANAGEMENT

Call For Papers	815
Case Study of Designing Interface of the AGH Students Information Bulletin Work Support System	817
<i>Natalia Nitarska, Piotr Wiśniewski, Krzysztof Kluza, Mateusz Zaremba, Antoni Ligeza</i>	
The impact of the multi-variant remote work model on knowledge management in enterprises. Applied tools.	827
<i>Anna Nowacka, Dorota Jelonek</i>	
Named Entity Recognition System for the Biomedical Domain	837
<i>Raghav Sharma, Deependra Singh, Raksha Sharma</i>	
Representing and Managing Experiential Knowledge with Decisional DNA and its Drimos® Extension	841
<i>Edward Szczerbicki, Cesar Sanin, Karina Sterling-Zuluaga</i>	

SOFTWARE, SYSTEM AND SERVICE ENGINEERING

Call For Papers	845
------------------------	------------

ADVANCES IN SOFTWARE, SYSTEM AND SERVICE ENGINEERING

Call For Papers	847
Extensible Conflict-Free Replicated Datatypes for Real-time Collaborative Software Engineering	849
<i>Istvan David, Eugene Syriani, Constantin Masson</i>	
Sensor Data Protection in Cyber-Physical Systems	855
<i>Anton Hristozov, Eric Matson, Eric Dietz, Marcus Rogers</i>	
Discovering interactions between applications with log analysis	861
<i>Lukasz Korzeniowski, Krzysztof Goczyla</i>	

JOINT 42ND IEEE SOFTWARE ENGINEERING WORKSHOP AND 9TH INTERNATIONAL WORKSHOP ON CYBER-PHYSICAL SYSTEMS

Call For Papers	871
Software Sentiment Analysis using Deep-learning Approach with Word-Embedding Techniques	873
<i>Venkata Krishna Chandra Mula, Lov Kumar, Lalita Bhanu Murthy, Aneesh Krishna</i>	
Scrum, Kanban or a Mix of Both? A Systematic Literature Review	883
<i>Necmettin Ozkan, Sevval Bal, Tugba Gurgen Erdogan, Mehmet Şahin Gök</i>	
Software Requirements Classification using Deep-learning Approach with Various Hidden Layers	895
<i>Sanidhya Vijayvargiya, Lov Kumar, Lalita Bhanu Murthy, Sanjay Misra</i>	

7TH WORKSHOP ON MODEL DRIVEN APPROACHES IN SYSTEM DEVELOPMENT

Call For Papers	905
Model-Based System Engineering Adoption in the Vehicular Systems Domain <i>Henrik Gustavsson, Jan Carlson, Eduard Enoiu</i>	907
An Experimentation Framework for Specification and Verification of Web Services <i>Szymon Kutra, Wiktor Daszczuk, Danny Czejdo</i>	913
Beyond Low-Code Development: Marrying Requirements Models and Knowledge Representations <i>Kamil Rybiński, Michał Smialek</i>	919

7TH DOCTORAL SYMPOSIUM ON RECENT ADVANCES IN INFORMATION TECHNOLOGY

Call For Papers	929
Impact of clustering unlabeled data on classification: case study in bipolar disorder <i>Olga Kamińska, Katarzyna Kaczmarek-Majer, Olgierd Hryniewicz</i>	931
Parameters Estimation of a Lotka-Volterra Model in an Application for Market Graphics Processing Units <i>Dzhakhongir Normatov, Paolo Mercorelli</i>	935
Tag and correct: high precision post-editing approach to speech recognition errors correction <i>Tomasz Ziętkiewicz</i>	939
Author Index	943

Two new operations over extended index matrices and their applications in Big Data

Krassimir Atanasov*[†], Veselina Bureva[†]
 *Dept. of Bioinformatics and Mathematical Modelling
 Institute of Biophysics and Biomedical Engineering,
 Bulgarian Academy of Sciences
 105 Acad. G. Bonchev Str., 1113 Sofia, Bulgaria
[†]Intelligent Systems Laboratory
 “Prof. Asen Zlatarov” University
 Burgas–8010, Bulgaria
 E-mail: krat@bas.bg, vbureva@btu.bg

Abstract—The Index Matrices (IMs) are extensions of the matrices of algebra. Over the IMs different operations, relations and operators are defined. In the present paper, two new operations over IMs are defined and some of their properties are studied. An application of these operations for describing of Big Data procedure is discussed. Hortonworks Data Platform (HDP) is used to provide capabilities for data warehouse processing in the Big Data environment. Apache Hive is selected for data warehouse construction and querying. Firstly, data warehouse for product sells is implemented. The tables for employees, customers, products and product sells are created. Thereafter the new index matrix operations for difference between two IMs are executed for the first time in the Big Data environment. SQL queries are written to demonstrate the operations. The new index matrix operations are executed using SQL JOIN notation and logical operator NOT EXISTS.

Index Terms—Big Data, Extended index matrix, Operation.

I. INTRODUCTION

THE CONCEPT of Index Matrix (IM) was introduced in 1984 and in more details - in 1987 [2], but the full description of the research over them was published in [3] exactly 30 years later.

Different extensions and modifications of the concept of an IM are described in [3]. One of them is an Extended IM (EIM), introduced firstly in [4]. They include as partial cases standard IM with elements of real or complex numbers, the (0, 1)-IM with elements from set {0, 1}, the logical IM with elements variables, propositions or predicates, the intuitionistic fuzzy IMs. The elements of the EIM can be each objects, in this number - whole IMs.

Different relations, operations and operators are defined over IMs and more general - over EIM. Only part of them have analogues in the theory of the standard matrices (see, e.g., [6], [7]).

Here, two new operations over EIMs are defined and some of their properties are studied. It will be obvious that these new operations can be transfer over each one of the partial cases of the EIM.

Firstly, we give the definition of an EIM.

Let \mathcal{I} be a fixed set of indices,

$$\mathcal{I}^n = \{(i_1, i_2, \dots, i_n) | (\forall j : 1 \leq j \leq n)(i_j \in \mathcal{I})\}$$

and

$$\mathcal{I}^* = \cup_{1 \leq n \leq \infty} \mathcal{I}^n.$$

Let \mathcal{X} be a fixed set of some objects. In the particular cases, they can be either real numbers, or only the numbers 0 or 1, or logical variables, propositions or predicates, etc.

Let operations $\circ, * : \mathcal{X} \times \mathcal{X} \rightarrow \mathcal{X}$ be fixed and let $(\mathcal{X}, \circ, e_\circ)$ and $(\mathcal{X}, *, e_*)$ be groups with unit elements e_\circ and e_* , respectively. For example, when operation “ \circ ” is “+” or “ \cdot ”, e_\circ will be 0, while when it is “ \times ” or “ \cdot ” – it will be 1. In some cases, it is suitable to define the unit element by \perp and it to be an empty object.

An EIM with index sets K, L ($K, L \subset \mathcal{I}^*$) and elements from set \mathcal{X} is called the object (see, [4], [3]):

$$[K, L, \{a_{k_i, l_j}\}] \equiv$$

		l_1	\dots	l_j	\dots	l_n
k_1	a_{k_1, l_1}	\dots	a_{k_1, l_j}	\dots	a_{k_1, l_n}	
\vdots	\vdots	\dots	\vdots	\dots	\vdots	
k_i	a_{k_i, l_1}	\dots	a_{k_i, l_j}	\dots	a_{k_i, l_n}	
\vdots	\vdots	\dots	\vdots	\dots	\vdots	
k_m	a_{k_m, l_1}	\dots	a_{k_m, l_j}	\dots	a_{k_m, l_n}	

where $K = \{k_1, k_2, \dots, k_m\}$, $L = \{l_1, l_2, \dots, l_n\}$, for $1 \leq i \leq m$, and $1 \leq j \leq n : a_{k_i, l_j} \in \mathcal{X}$.

In [3], for the IMs $A = [K, L, \{a_{k_i, l_j}\}]$, $B = [P, Q, \{b_{p_r, q_s}\}]$, operations that are analogous to the usual matrix operations of addition and multiplication are defined, as well as other, specific ones. Here, we give only three of these operations, that will be used below.

Addition

$$A \oplus_{(\circ)} B = [K \cup P, L \cup Q, \{c_{t_u, v_w}\}],$$

where

$$c_{t_u, v_w} = \begin{cases} a_{k_i, l_j}, & \text{if } t_u = k_i \in K \text{ and } v_w = l_j \in L - Q \\ & \text{or } t_u = k_i \in K - P \text{ and } v_w = l_j \in L; \\ b_{p_r, q_s}, & \text{if } t_u = p_r \in P \text{ and } v_w = q_s \in Q - L \\ & \text{or } t_u = p_r \in P - K \text{ and } v_w = q_s \in Q; \\ a_{k_i, l_j} \circ b_{p_r, q_s}, & \text{if } t_u = k_i = p_r \in K \cap P \\ & \text{and } v_w = l_j = q_s \in L \cap Q \\ 0, & \text{otherwise} \end{cases}$$

Structural subtraction

$$A \ominus B = [K - P, L - Q, \{c_{t_u, v_w}\}],$$

where here and below “ $-$ ” is the set-theoretic difference operation and

$$c_{t_u, v_w} = a_{k_i, l_j}, \text{ for } t_u = k_i \in K - P \text{ and } v_w = l_j \in L - Q.$$

Projection

$$pr_{M, N} A = [M, N, \{b_{k_i, l_j}\}],$$

where $M \subseteq K$, $N \subseteq L$, and for each $k_i \in M$ and each $l_j \in N$, $b_{k_i, l_j} = a_{k_i, l_j}$.

II. DEFINITIONS OF THE TWO NEW OPERATIONS

Let the two EIMs $A = [K, L, \{a_{k_i, l_j}\}]$ and $B = [P, Q, \{b_{p_r, q_s}\}]$ are given.

The first, simpler, operation is defined by

$$A \overset{\bullet}{\underset{1}{\circ}} B = [K \div P, L \div Q, \{c_{u_v, w_t}\}],$$

where and below for every two arbitrary sets X, Y :

$$X \div Y = (X - Y) \cup (Y - X);$$

$$c_{u_v, w_t} = \begin{cases} a_{k_i, l_j}, & \text{if } u_v = k_i \in K - P \text{ and } w_t = l_j \in L - Q \\ b_{p_r, q_s}, & \text{if } w_t = p_r \in P - K \text{ and } w_t = q_s \in Q - L \\ \perp & \text{otherwise} \end{cases}$$

The second operation is defined by

$$A \overset{\bullet}{\underset{2}{\circ}} B = [K \cup P, L \cup Q, \{c_{u_v, w_t}\}],$$

where

$$c_{u_v, w_t} = \begin{cases} a_{k_i, l_j}, & \text{if } u_v = k_i \in K - P \text{ and } w_t = l_j \in L \\ & \text{or } u_v = k_i \in K \text{ and } w_t = l_j \in L - Q \\ b_{p_r, q_s}, & \text{if } w_t = p_r \in P - K \text{ and } w_t = q_s \in Q \\ & \text{if } w_t = p_r \in P \text{ and } w_t = q_s \in Q - L \\ \perp & \text{otherwise} \end{cases}$$

The geometrical interpretations of both operations are shown on Figures 1 and 2.

From the definitions and geometrical interpretations of both operations we see immediately that the following assertions are valid.

Theorem 1. For every two EIMs A and B , for each operation \circ , and for operation $*$ defined for every two $x, y \in \mathcal{X}$ by $x * y = e_*$ there are follows:

$$A \overset{\bullet}{\underset{1}{\circ}} B = (A \ominus B) \oplus_{(\circ)} (B \ominus A),$$

$$A \overset{\bullet}{\underset{2}{\circ}} B = A \oplus_{(*)} B.$$

Proof. For the definitions of the operations \ominus and \oplus_{\circ} we obtain

$$\begin{aligned} (A \ominus B) \oplus_{(\circ)} (B \ominus A) &= [K - P, L - Q, \{c_{t_u, v_w}\}] \oplus_{(\circ)} [P - K, Q - L, \{d_{e_f, g_h}\}] \\ &= [(K - P) \cup (P - K), (L - Q) \cup (Q - L), \{c'_{t'_{u'}, v'_{w'}}\}] \\ &= [K \div P, L \div Q, \{c'_{t'_{u'}, v'_{w'}}\}] = A \div B, \end{aligned}$$

because

$$c'_{t'_{u'}, v'_{w'}} = \begin{cases} c_{t_u, v_w} = a_{k_i, l_j}, & \text{for } t'_{u'} = t_u = k_i \in K - P \text{ and } v'_{w'} = v_w \\ d_{e_f, g_h} = b_{p_r, q_s}, & \text{for } t'_{u'} = e_f = p_r \in P - K \text{ and } v'_{w'} = g_h \end{cases}$$

Obviously, operation \circ cannot be applied over the elements of both EIMs because there are not at least two elements from the both EIMs that have equal indices.

The second equality is proved by a similar way. \square

Theorem 2. For every two EIMs A and B , for each operation \circ :

$$A \overset{\bullet}{\underset{1}{\circ}} B = pr_{K-P, L-Q} A \oplus_{(\circ)} pr_{P-K, Q-L} B,$$

$$\begin{aligned} A \overset{\bullet}{\underset{2}{\circ}} B &= pr_{K-P, L} A \oplus_{(\circ)} pr_{K \cap P, L \div Q} A \oplus_{(\circ)} pr_{P-K, Q} A \\ &= pr_{K, L-Q} A \oplus_{(\circ)} pr_{K \div P, L \cap Q} A \oplus_{(\circ)} pr_{P, Q-L} A. \end{aligned}$$

The proof is similar to the above one.

Let \mathcal{M} be the set of all EIMs with elements from \mathcal{X}^1 .

Let

$$I_{\emptyset} = [\emptyset, \emptyset, \{\perp\}].$$

Theorem 3. $(\mathcal{M}, \overset{\bullet}{\underset{1}{\circ}}, I_{\emptyset})$ is a commutative group.

Proof. From the definition of operation “ $\overset{\bullet}{\underset{1}{\circ}}$ ” it follows directly that for each two EIMs $A, B \in \mathcal{M}$, $A \overset{\bullet}{\underset{1}{\circ}} B \in \mathcal{M}$.

Using the well-known equality for every three sets X, Y and Z :

$$(X \div Y) \div Z = X \div (Y \div Z),$$

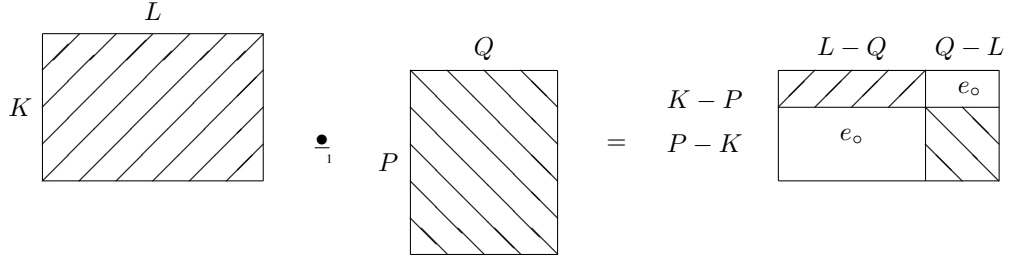
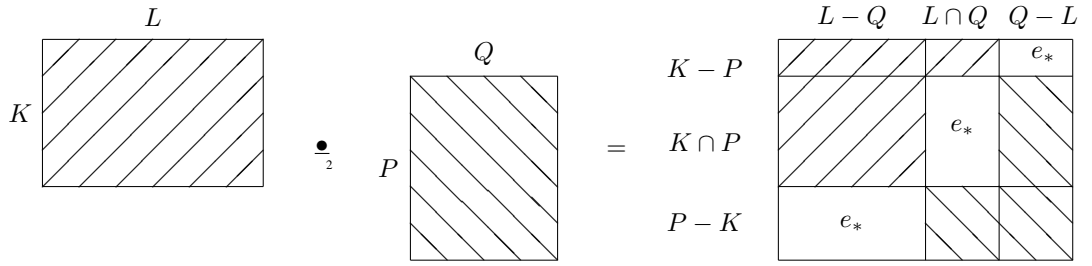
for every three EIMs $A, B, C \in \mathcal{M}$, where

$$C = [D, E, \{c_{d_f, e_g}\}]$$

we obtain

$$\begin{aligned} (A \overset{\bullet}{\underset{1}{\circ}} B) \overset{\bullet}{\underset{1}{\circ}} C &= ([K, L, \{a_{k_i, l_j}\}] \overset{\bullet}{\underset{1}{\circ}} [P, Q, \{b_{p_r, q_s}\}]) \overset{\bullet}{\underset{1}{\circ}} [D, E, \{c_{d_f, e_g}\}] \\ &= [K \div P, L \div Q, \{x_{u_v, w_t}\}] \overset{\bullet}{\underset{1}{\circ}} [D, E, \{c_{d_f, e_g}\}] \end{aligned}$$

¹When \mathcal{X} is a set (class) in the sense of NBG-set theory of all predicates, then \mathcal{M} will be a set (class).

Fig. 1. Geometrical interpretation of operation $\dot{\bullet}_1$ Fig. 2. Geometrical interpretation of operation $\dot{\bullet}_2$

(where each element x_{u_v, w_t} is some element a_{k_i, l_j} or some element b_{p_r, q_s})

$$= [(K \div P) \div D, (L \div Q) \div E, \{y_{\alpha\beta, \gamma\delta}\}]$$

(where each element $y_{\alpha\beta, \gamma\delta}$ is some element a_{k_i, l_j} or some element b_{p_r, q_s}), or some element c_{d_f, e_g})

$$= [K \div (P \div D), L \div (Q \div E), \{y_{\alpha\beta, \gamma\delta}\}]$$

$$= ([K, L, \{a_{k_i, l_j}\}] \dot{\bullet}_1 ([P \div D, Q \div E, \{z_{\varepsilon\zeta, \eta\theta}\}]))$$

(where each element $z_{\varepsilon\zeta, \eta\theta}$ is some element b_{p_r, q_s}), or some element c_{d_f, e_g})

$$= [K, L, \{a_{k_i, l_j}\}] \dot{\bullet}_1 ([P, Q, \{b_{p_r, q_s}\}]) \dot{\bullet}_1 [D, E, \{c_{d_f, e_g}\}].$$

Now, for EIM I_\emptyset we obtain

$$A \dot{\bullet}_1 I_\emptyset = [K \div \emptyset, L \div \emptyset, \{c_{u_v, w_t}\}]$$

(where each element c_{u_v, w_t} coincides with the element a_{k_i, l_j} from A for $u_v = k_i, w_t = l_j$)

$$= [K, L, \{a_{k_i, l_j}\}] = A.$$

Analogously is checked that

$$I_\emptyset \dot{\bullet}_1 A = A.$$

From the well-known equality $X \div Y = Y \div X$ it follows that

$$A \dot{\bullet}_1 B = [K \div P, L \div Q, \{c_{u_v, w_t}\}]$$

$$m = [P \div K, Q \div L, \{c_{u_v, w_t}\}] = B \dot{\bullet}_1 A,$$

i.e., the operation $\dot{\bullet}_1$ is commutative.

Finally,

$$A \dot{\bullet}_1 A = [K \div K, L \div L, \{x_{u_v, w_t}\}] = [\emptyset, \emptyset, \{x_{u_v, w_t}\}] = I_\emptyset,$$

because of lack of indices, element x_{u_v, w_t} must be \perp . \square

Theorem 4. $(\mathcal{M}, \dot{\bullet}_2, I)$ is a commutative monoid.

The proof is similar to this of Theorem 3, without the last part, i.e., as above, we can check that $(\mathcal{M}, \dot{\bullet}_2, I)$ is a commutative monoid, but it is not a group because the fact that there is not a set Y that for some non-empty set X to be valid $X \cup Y = \emptyset$.

III. AN EXAMPLE OF DIFFERENCE OPERATIONS IN THE BIG DATA ENVIRONMENT

Hortonworks Data Platform (HDP) is an open source framework for distributed storage and processing of huge datasets retrieving from different sources. HDP is used to discover insights from structured and unstructured data in the cloud or on-premises. It includes Big Data tools as Hadoop,

The screenshot shows the Ambari web interface for Hive. The top navigation bar includes 'Ambari', 'Sandbox', 'ops', 'alerts', 'Dashboard', 'Services', 'Hosts', 'Alerts', and 'Admin'. The main header is 'HIVE' with a '+ New' button. Below the header, there are tabs for 'QUERY', 'JOBS', 'TABLES', 'SAVED QUERIES', 'UDFs', and 'SETTINGS'. The 'TABLES' tab is active, showing a search bar with 'bigdatadb' and a 'Browse' button. Below the search bar, there are icons for 'customers', 'employees', 'productsales', and 'products'. The 'productsales' table is selected, and its structure is displayed in a table format:

COLUMN NAME	COLUMN TYPE	COMMENT
sn	int	
employeeen	int	
customern	int	
productn	int	
quantity	int	

Fig. 3. Data warehouse bigdatadb view in Apache Hive environment

Yarn, MapReduce, Hbase, Hive, Flume, Kafka, Druid [5]. The example of the difference operations is performed in the environment of Apache Hive. Apache Hive is a data warehouse used for reading, writing, and managing large amounts of data stored in distributed storage in SQL. Apache Hive flexibility can be extended using the used-defined functions (UDF) [1]. In the current investigation a data warehouse for product sells is implemented. The tables are uploaded using previously prepared csv files. Apache Hive supports only the sets operations *Union All* and *Union [Distinct]*. The *Intersect* and *Except (Minus)* operations are not included. In the current investigation for the first time the new index matrix operations performing operation difference will be applied by the analogy of the SQL *JOIN* clause and the logical operator *NOT EXISTS* in the Big Data environment. The tables *Customers* and *Employees* from *bigdatadb* data warehouse (Fig. 3). The authors will not compare the indexed field Number – the comparison will be performed using the columns for the first name and the last name of the tables. There are from the same data type. The tables *Employees* (Fig. 4) and *Customers* (Fig. 5) have the following from:

An example of the standard JOIN operation has the following form:

```
SELECT FName, Lname, CFname, CLName
FROM Employees JOIN ProductSales
ON Employees.Number=ProductSales.EmployeeN
JOIN Customers
ON Customers.CNumber=ProductSales.CustomerN
```

employees.number	employees.fname	employees.lname	employees.town
1	Ivan	Dimitrov	Burgas
2	George	Radev	Burgas
3	Simona	Ivanova	Sofia
4	Radi	Hristov	Burgas
5	Vasilena	Moneva	Sofia
6	Hristo	Rachev	Burgas
7	Qni	Simov	Burgas
8	Radina	Tomova	Sofia
9	Stanislav	Stoyanov	Sofia
10	Radostina	Dimitrova	Burgas
11	Dimo	Dimov	Burgas
12	Maria	Yaneva	Burgas
13	Dimitar	Stoilov	Sofia
14	Svetla	Iliyanova	Burgas
15	Margarita	Simova	Burgas
16	Radomira	Kirova	Sofia
17	Radovesta	Todorova	Sofia
18	Valeria	Taneva	Sofia
19	Silviya	Stancheva	Burgas
20	Stilyan	Stoilov	Burgas

Fig. 4. Table of Employees

customers.cnumber	customers.cfname	customers.clname	customers.ctown
1	Ivan	Dimitrov	Burgas
2	George	Radev	Burgas
3	Siyana	Tumova	Sofia
4	Radina	Mihaylova	Sofia
5	Stancho	Dimitrov	Burgas
6	Hari	Dimov	Burgas
7	Qni	Simov	Burgas
8	Radina	Tomova	Sofia
9	Mihail	Radev	Burgas
10	Hrisi	Staneva	Sofia
11	Dimo	Dimov	Burgas
12	Maria	Yaneva	Burgas
13	Monika	Ganeva	Burgas
14	Stanimit	Ganeva	Sofia
15	Hristo	ivanov	Sofia
16	Iliyan	Fotev	Sofia
17	Hristiana	Racheva	Burgas
18	Valeria	Taneva	Sofia
19	Silviya	Stancheva	Burgas
20	Stilyan	Stoilov	Burgas

Fig. 5. Table of Customers

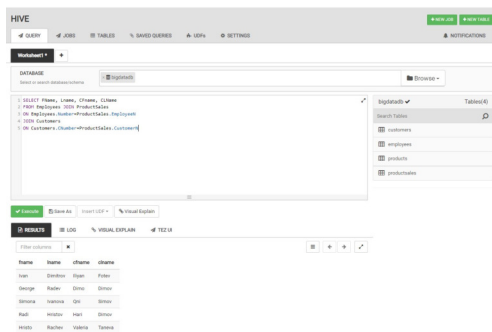


Fig. 6. Execution of standard JOIN operation

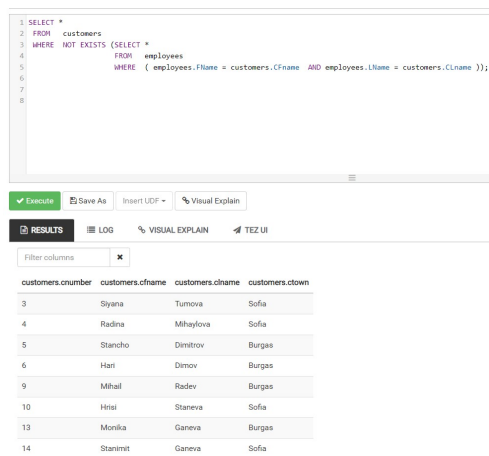


Fig. 7. SQL query for difference operation – case 1

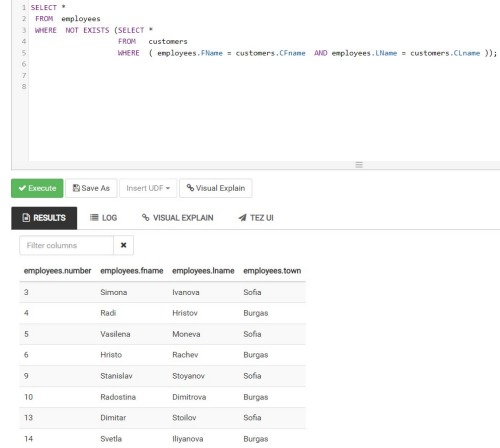


Fig. 8. SQL query for difference operation – case 2

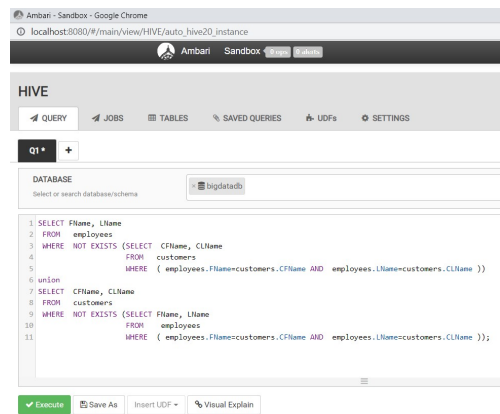


Fig. 9. SQL query for difference operation – case 3

The result presents the names of the customers and the employees. The SQL query in the Apache Hive environment is presented on the Fig. 6.

The simulation of difference operation is performed using the *NOT EXIST* logical operator in the *SELECT* statement and comparison of the desired columns from the tables in the sub-query (Fig. 7).

```
SELECT * FROM customers
WHERE NOT EXISTS (SELECT *
FROM employees
WHERE (employees.FName = customers.CFname
AND employees.LName = customers.CLname)
);
```

The result of the query presents customers not included in the list of the employees. The columns for first names and last names are compared.

The same query with replaced tables in the query and sub-query can be executed to receive the employees not included in the table of the customers (Fig. 8).

The result of the query is presented on the Fig. 10. It combines the employees that are not customers and the customers that are not employees.

The combination of the previous queries using the UNION operator is executed (Fig. 9).

RESULTS		LOG	VISUAL EXPLAIN
Filter columns			
_u2.fname	_u2.lname		
Dimitar	Stoilov		
Hari	Dimov		
Hrisi	Staneva		
Hristiana	Racheva		
Hristo	Rachev		
Hristo	ivanov		
Iliyan	Fotev		
Margarita	Simova		
Mihail	Radev		
Monika	Ganeva		
Radi	Hristov		
Radina	Mihaylova		
Radomira	Kirova		
Radostina	Dimitrova		
Radovesta	Todorova		
Simona	Ivanova		
Siyana	Tumova		
Stancho	Dimitrov		
Stanimit	Ganeva		
Stanislav	Stoyanov		
Svetla	Iliyanova		
Vasilena	Moneva		

Fig. 10. Result of SQL query for difference operation – case 3

```

SELECT F Name, LName
FROM employees
WHERE NOT EXISTS (SELECT CFName, CLName
FROM customers
WHERE (employees.FName=customers.CFName
AND employees.LName=customers.CLName )
)
UNION
SELECT CFName, CLName
FROM customers
WHERE NOT EXISTS (SELECT FName, LName
FROM employees
WHERE (employees.FName=customers.CFName
AND employees.LName=customers.CLName )
);

```

IV. CONCLUSION

Two new operations are introduced in the paper. In future, other properties of them will be studied and some other applications will be discussed. A part of them will be related to Big Data and Data Mining as continuation of the discussed in the present paper.

ACKNOWLEDGMENT

The authors acknowledge the support from the project UNITE BG05M2OP001-1.001-0004 /28. 02.2018 (2018-2023).

REFERENCES

- [1] Apache Hive, <https://hive.apache.org/>
- [2] Atanassov K. Generalized index matrices, Comptes rendus de l'Academie Bulgare des Sciences, Vol. 40, 1987, No. 11, 15–18.
- [3] Atanassov, K., Index Matrices: Towards an Augmented Matrix Calculus, Springer, Cham, 2014.
- [4] Atanassov, K., Extended index matrices, Proc. of the 7-th IEEE Conference "Intelligent Systems", Warsaw, 24-26 Sept. 2014 (P. Angelov, K. Atanassov, L. Doukowska, M. Hadjiski, V. Jotsov, J. Kacprzyk, N. Kasabov, S. Sotirov, E. Szmids, S. Zadrozny, Eds.), Springer, Cham, 2015, 305-317.
- [5] Hortonworks Data platform, <https://www.cloudera.com/products/hdp.html>
- [6] Lankaster, P. Theory of Matrices, Academic Press, New York, 1969.
- [7] Zhang, F. Matrix Theory. Springer, New York, 2011.

Hybridization of Fuzzy Sets and Rough Sets: Achievements and Opportunities

Chris Cornelis

Computational Web Intelligence

Department of Applied Mathematics, Computer Science and Statistics

Ghent University

Krijgslaan 281 (S9), B-9000 Gent, Belgium

Email: Chris.Cornelis@UGent.be

Abstract—Fuzzy rough sets are the fruit of an intense and long-lasting collaboration effort between fuzzy set theory and rough set theory. Seminal research on the hybridization originated in the late 1980’s, and has inspired generations of researchers from around the globe to address both theoretical and practical challenges. In this paper, we gauge the state-of-the-art in this domain and identify opportunities for further development. In particular, we highlight the potential of fuzzy quantifiers in creating new robust fuzzy rough models, we advocate closer integration with granular computing as a stepping stone for designing rule induction algorithms, and we contemplate the role of fuzzy rough sets vis-à-vis explainable artificial intelligence.

I. INTRODUCTION

FUZZY ROUGH SETS emerge as a combination of fuzzy sets (Zadeh [62], 1965) and rough sets (Pawlak [43], 1982): while the former model vague information by recognizing that membership to certain concepts, or logical truth of certain propositions, is a matter of degree, the latter handle potentially inconsistent information by providing a lower and upper approximation of a concept, using the equivalence classes of an indiscernibility relation as building blocks. Both frameworks can be integrated from at least three different perspectives:

- 1) Concepts may be fuzzy rather than exact, allowing that objects belong to it to varying degrees. For example, in a data set containing information about hotels, one may be interested to characterize the concept of hotels considered as “expensive”, an inherently vague predicate. Then, each hotel’s membership to the fuzzy concept “expensive hotel” will be expressed using a value between 0 (not belonging to the concept at all, i.e., not expensive) and 1 (fully meeting the concept’s membership conditions, i.e., definitely expensive).
- 2) The indiscernibility relation, expressing that objects may or may not be distinguished from each other, may be gradual rather than strict; this reflects the intuitive idea that some objects are more similar to each other than others, and therefore they should be related to a higher degree (again a value between 0 and 1).

- 3) The condition for belonging to the lower and upper approximation may be expressed using *fuzzy quantifiers*. For example, classically an object is a member of the lower approximation of a concept if *all* objects indiscernible from it also belong to the concept. Here, instead of the traditional universal quantifier (\forall), one may use a fuzzy quantifier like “most”. The purpose of such a relaxation is to introduce a measure of tolerance towards inconsistency into the approximations, making them more robust.

The first two principles were already present early on in the work of Fariñas del Cerro and Prade [17], and Dubois and Prade [16], while the third one was first explored in 2007 by the theory of vaguely quantified rough sets [5] and recently revived by the introduction of Choquet-based fuzzy rough sets [49]. In this paper, we want to highlight the potential of fuzzy quantifiers in designing robust and interpretable fuzzy rough set models, which are in particular relevant for applications in data analysis [52]. The latter are becoming more and more widespread, and include, amongst others, feature selection [8], [13], [39], instance selection [29], [50], [51], instance-based classification [25], [30], [32], [33], cognitive networks [37], imbalanced classification [47], [54], multi-instance classification [53] and multi-label classification [35], [55].

Given that many of the aforementioned applications are instance-based or greedy approaches, whereas most successful applications of the rough set paradigm are rule based systems (see e.g. [23], [24]), it may be argued that the full potential of the hybrid theory has not yet been tapped (some notable exceptions are [22], [65], [66]), taking into account also the vast body of existing research on fuzzy rule based systems [2], [18], [26]. An important key to this logical next step lies in the field of granular computing [4], [59], an information processing paradigm centered on the segmentation of complex information into smaller pieces called information granules. Both rough sets and fuzzy sets relate to granular computing; it is well-known that the lower and upper approximation can be represented as unions of simple sets or granules induced from data [60], while Zadeh [64] identified fuzziness as a key part of the granulation in human cognition, and with

This work was supported by the Odysseus programme (grant number G0H9118N) of the Research Foundation – Flanders (FWO).

the help of fuzzy granules, a fuzzy rule based system may be set up. An important advantage of fuzzy rules, and of fuzzy logic in general, is that they allow for explanations using linguistic expressions. This property can be utilized for the development of interpretable machine learning algorithms, a research direction which is currently attracting a lot of researchers' attention [36].

The remainder of this paper is structured as follows: in Section II, we recall important preliminaries from both rough set and fuzzy set theory, while in Section III, we outline the main steps and results of the hybridization process. In Section IV, we pay attention to robust fuzzy rough set models, which play an important role for practical applications of the theory. Finally, in Section V, we offer an informal discussion of ongoing challenges and new opportunities for the hybrid theory.

II. PRELIMINARIES: FUZZY SETS AND ROUGH SETS

A. Rough sets

We first recall Pawlak's definition [43] of rough sets, which is also called the Indiscernibility-based Rough Set Approach (IRSA).

Definition 2.1: Let U be a set of objects and E an equivalence relation expressing indiscernibility, i.e., E is

- reflexive: $(u, u) \in E$,
- symmetric: if $(u, v) \in E$ then $(v, u) \in E$,
- transitive: if $(u, v) \in E$ and $(v, w) \in E$, then also $(u, w) \in E$.

For any $A \subseteq U$, the *lower and upper approximation* of A are defined as:

$$\underline{\text{apr}}_E(A) = \{u \in U : [u]_E \subseteq A\} \quad (1)$$

$$\overline{\text{apr}}_E(A) = \{u \in U : [u]_E \cap A \neq \emptyset\} \quad (2)$$

The couple $(\underline{\text{apr}}_E(A), \overline{\text{apr}}_E(A))$ is called the *rough set* of A . The equations (1) and (2) can be expressed equivalently using logical operators: for $u \in U$,

$$u \in \underline{\text{apr}}_E(A) \Leftrightarrow (\forall v \in U)((v, u) \in E \Rightarrow v \in A) \quad (3)$$

$$u \in \overline{\text{apr}}_E(A) \Leftrightarrow (\exists v \in U)((v, u) \in E \wedge v \in A) \quad (4)$$

It is also easily verified that $\underline{\text{apr}}_E(A) \subseteq A \subseteq \overline{\text{apr}}_E(A)$, which justifies the terms "lower and upper approximation". Moreover, the rough set approximations satisfy various other properties, for example set monotonicity:

$$A \subseteq A' \Rightarrow \underline{\text{apr}}_E(A) \subseteq \underline{\text{apr}}_E(A') \wedge \overline{\text{apr}}_E(A) \subseteq \overline{\text{apr}}_E(A') \quad (5)$$

which expresses that if a concept becomes larger, its approximations naturally should not decrease. On the other hand, relation monotonicity:

$$E \subseteq E' \Rightarrow \underline{\text{apr}}_E(A) \supseteq \underline{\text{apr}}_{E'}(A) \wedge \overline{\text{apr}}_E(A) \subseteq \overline{\text{apr}}_{E'}(A) \quad (6)$$

states that when equivalence classes become larger (more objects are indiscernible from each other), the lower approximation gets smaller, while more objects populate the upper approximation.

In case $\underline{\text{apr}}_E(A) = \overline{\text{apr}}_E(A)$, we call A an *exact set*. An equivalent way of expressing that A is exact is

$$A = \bigcup_{u \in A} [u]_E \quad (7)$$

In other words, A can be seen as a union of basic building blocks or granules, which correspond to equivalence classes of E . We call (7) the *granular representation* of A , and A is also called a *granularly representable set*. The following proposition highlights the special role of the lower and upper approximation as specific exact sets.

Proposition 2.2: For $A \subseteq U$, the greatest granularly representable set that is included in A is equal to $\underline{\text{apr}}_E(A)$, while the smallest granularly representable set that includes A is equal to $\overline{\text{apr}}_E(A)$.

On the other hand, the granular representation is also closely connected with the notion of consistency.

Proposition 2.3: Set $A \subseteq U$ is granularly representable if and only if it satisfies the consistency property, i.e., iff

$$(\forall u, v \in U)((v, u) \in E \wedge u \in A \Rightarrow v \in A) \quad (8)$$

Consistency expresses that if two objects are indiscernible and one of them belongs to a given concept A , the second object should necessarily also be part of the concept. This property is desirable in classification problems, where the goal is to establish meaningful patterns that allow to decide the membership of unseen objects to given decision classes. In this context, objects are also called instances and are characterized by their values for a number of attributes from a set \mathcal{A} . The domain of every attribute $a \in \mathcal{A}$ consists of a finite number of nominal values, and every instance $u \in U$ takes one of those values denoted with $a(u)$. Then, the equivalence relation E is constructed as

$$(u, v) \in E \Leftrightarrow (\forall a \in \mathcal{A})(a(u) = a(v)) \quad (9)$$

The granular representation of rough sets is in particular very useful from the perspective of rule induction. The problem of rule induction for classification tasks amounts to generating a set of rules which relate descriptions of objects by subsets of attributes with particular decision classes. Basic granules, from which rough sets are composed, can be interpreted as human readable "if..., then..." rules, and can be used to construct a rule based inference system as a prediction model. Well-known examples of rule induction algorithms are LEM2 [23] and MODLEM [24].

Pawlak's theory has been generalized in various different ways. For example, dropping the symmetry requirement from E leads to the Preorder-based Rough Set Approach (PRSA, [40]), which contains as a special case the Dominance-based Rough Set Approach (DRSA, [21]). In the latter, the domain of attributes now contains ordinal values, and the indiscernibility relation is replaced by a dominance relation. For clarity and brevity of the exposition, in the remainder of this paper we will focus on the indiscernibility-based approach, although many of the presented results remain valid for more general settings.

B. Fuzzy sets

Given a universal set U , Zadeh defined a fuzzy set A in U simply as a mapping from U to the unit interval $[0, 1]$, where $A(u)$ is called the membership degree of object u to A . It expresses to what extent u satisfies the vague property expressed by the fuzzy set A . For example, if U is a set of hotels from a given area, we may evaluate their expensiveness, based on their quoted nightly rate for a double room, as a fuzzy set A in U . Clearly, the assignment of membership degrees is both subjective and context-dependent, as it would depend for example on the budget of the person making the booking, and the area where the search is performed. However, an intuitive constraint in this case would be that the higher the quoted rate, the larger the membership degree should be. In this example, as in most practical applications of fuzzy set theory, there is an underlying numerical scale (a subset of the real numbers) on which the evaluation is made, and the ordering on that scale constrains the assignment of membership degrees.

In a similar vein, a binary fuzzy relation R in U is defined as a fuzzy set in U^2 , i.e., for any two objects u and v in U , $R(u, v)$ expresses the degree to which they relate. Fuzzy relations may be used for example to generalize the equivalence relation E from Section II-A, to establish to what extent two objects are *similar* (as opposed to a black-or-white assessment whether they are indiscernible or not). Such a fuzzy relation R should be at least reflexive and symmetric, i.e.,

$$R(u, u) = 1 \quad (10)$$

$$R(u, v) = R(v, u) \quad (11)$$

should hold for any u and v in U . In order to accommodate for the transitivity property, we first need an extension of the classical conjunction operator \wedge .

Definition 2.4: A triangular norm, or shortly t-norm, is a mapping $T : [0, 1]^2 \rightarrow [0, 1]$ that is commutative, associative, increasing in both arguments, and that satisfies the boundary condition $T(1, x) = x$ for all $x \in [0, 1]$.

Well-known representatives of the class of t-norms include the minimum, the product, and the Łukasiewicz t-norm defined by $T_{\mathbb{L}}(x, y) = \max(0, x + y - 1)$ for x, y in $[0, 1]$. The choice for a particular t-norm depends on the particular properties that one is interested in; for a comprehensive overview, we refer to [31].

Using a t-norm, we may now impose a kind of transitivity on fuzzy relations, and therefore extend the notion of an equivalence relation.

Definition 2.5: Let T be a t-norm. A fuzzy relation R in U that is reflexive, symmetric and satisfies

$$T(R(u, v), R(v, w)) \leq R(u, w) \quad (12)$$

for any u, v, w in U is called a fuzzy T -equivalence relation. Apart from logical conjunction, we will also require an extension of the boolean implication operator \Rightarrow .

Definition 2.6: An implicator is a mapping $I : [0, 1]^2 \rightarrow [0, 1]$ that is decreasing in its first argument and increasing

in its second one, and that satisfies the boundary conditions $I(0, 0) = I(0, 1) = I(1, 1) = 1$ and $I(1, 0) = 0$.

There exist numerous ways to construct implicators. Again, a detailed overview is out of the scope of this paper, and may be found in e.g. [3]. A popular approach is to associate implicators to t-norms by means of *residuation*, leading to the following definition of residuated implicators, or shortly R-implicators.

Definition 2.7: The R-implicator I_T associated to a t-norm T is defined by, for x, y in $[0, 1]$:

$$I_T(x, y) = \sup\{z \in [0, 1] \mid T(x, z) \leq y\} \quad (13)$$

As an example, the R-implicator associated to the Łukasiewicz t-norm can be obtained as $I_{T_{\mathbb{L}}}(x, y) = \min(1, 1 - x + y)$.

Finally, we recall that subsethood for fuzzy sets is defined as follows [62]: for fuzzy sets A and B in U ,

$$A \subseteq B \Leftrightarrow (\forall u \in U)(A(u) \leq B(u)) \quad (14)$$

III. HYBRIDIZATION: GENERAL FUZZY ROUGH SET MODEL

The equations (3) and (4) for determining membership to the classical lower and upper approximations can be “fuzzified” by making use of fuzzy logical connectives. This leads to the following definition [48], [6], [11].

Definition 3.1: Let A be a fuzzy set in U , R a fuzzy relation in U , I an implicator and T a t-norm. The lower and upper approximation of A are defined as, for $u \in U$,

$$\underline{\text{apr}}_R(A)(u) = \inf_{v \in U} I(R(v, u), A(u)) \quad (15)$$

$$\overline{\text{apr}}_R(A)(u) = \sup_{v \in U} T(R(v, u), A(u)) \quad (16)$$

The couple $(\underline{\text{apr}}_R(A), \overline{\text{apr}}_R(A))$ is called the *fuzzy rough set* of A . If $\underline{\text{apr}}_R(A) = A = \overline{\text{apr}}_R(A)$, A is called an exact fuzzy set.

Take note how these definitions extend their classical counterparts:

- 1) Object u belongs to the lower approximation of A to the extent that *for all* objects v , *if* v is related to u by R , *then* v should belong to A .
- 2) Object u belongs to the upper approximation of A to the extent that *there exists* an object v , such that *if* v is related to u by R , *and* v belongs to A .

In other words, the inf and sup operators naturally represent the \forall and \exists quantifier from Eq. (3) and (4), respectively, while the implicator I and t-norm T fulfil the role of the logical implication \Rightarrow and conjunction \wedge . When A is a classical, non-fuzzy set and R is a crisp equivalence relation, we again obtain Pawlak’s model. Depending on the specific choice of fuzzy connectives I and T , and the fuzzy relation R , some properties of this original model may or may not be preserved (see [12] for more details).

An important question is whether exact fuzzy sets possess a granular representation analogous to Eq. (7), as it would allow the above approximations to be used for generating fuzzy rules in a similar way as is done with crisp granules. Degang et al. [14] were the first to address this issue by formalizing the

notions of a fuzzy granule and granular representability of fuzzy sets.

Definition 3.2: Let R be a fuzzy T -equivalence relation in U for a given t-norm T and $\lambda \in [0, 1]$. The fuzzy granule corresponding to R , λ and T is the fuzzy set R_λ in U , defined by

$$R_\lambda(u) = \{(v, T(R(v, u), \lambda)); v \in U\} \quad (17)$$

We call a fuzzy set A in U granularly representable if

$$A = \bigcup \{R_{A(u)}(u); u \in U\} \quad (18)$$

In other words, A is granularly representable if it is the union of fuzzy granules $R_\lambda(u)$, where $\lambda = A(u)$ for each object u . The following proposition reveals that for particular choices of I and T , exact fuzzy sets indeed correspond to granularly representable ones, and vice versa.

Proposition 3.3: Let A be a fuzzy set in U , T a left-continuous t-norm and I its R-implicator. Then A is exact if and only if it is granularly representable.

Along the same lines, we can also generalize Proposition 2.2 and 2.3.

Proposition 3.4: For a fuzzy set A in U and a fuzzy T -equivalence relation, the greatest granularly representable fuzzy set that is included in A is equal to $\underline{\text{apr}}_R(A)$, while the smallest granularly representable set that includes A is equal to $\overline{\text{apr}}_R(A)$.

Proposition 3.5: Let R be a fuzzy T -equivalence relation. Fuzzy set A in U is granularly representable if and only if it satisfies the fuzzy consistency property, i.e., iff

$$(\forall u, v \in U)(T(R(v, u), A(u)) \leq A(v))$$

Note how fuzzy consistency provides us with a softened version of Eq. (8): the more similar u and v are, and the higher u 's membership to A , the more v should also belong to A .

IV. ROBUST FUZZY ROUGH SETS

The model of fuzzy rough sets described in the previous section offers considerable strength and flexibility, and lends itself very well for handling datasets with real-valued attributes, where fuzzy T -equivalence relations can be constructed by taking into account the distance between individual instances' attribute values. However, it may still be too rigid when applied in practical problems of data analysis, due to the occurrence of outliers. By the latter, we mean instances that do not follow the general data distribution, and which may negatively impact the quality of the fuzzy-rough approximations. In extreme cases, the lower approximation of a concept may be empty, while its upper approximation may contain all instances fully.

The root of the problem lies in the use of the inf and sup operators which, as we explained, correspond to the \forall and \exists quantifier, respectively. Because of this, an instance u will be fully excluded from $\underline{\text{apr}}_R(A)$ as soon as there exists another instance v such that $R(v, u) = 1$ and $A(v) = 0$, while on the other hand u will fully belong to $\overline{\text{apr}}_R(A)$ when an object

v can be found such that $R(v, u) = 1$ and $A(v) = 1$. This will occur independently of the choice of the implicator I and the t-norm T . While this effect may be mitigated by a thoughtful choice of the fuzzy relation R , it cannot be ruled out altogether as (partial) inconsistencies are commonplace in real applications.

In classical rough set theory, researchers also faced this problem, leading to probabilistic approaches like Ziarko's Variable Precision Rough Set (VPRS) model [67]. The latter relaxes Eq. (3) and (4) into

$$u \in \underline{\text{apr}}_E^p(A) \Leftrightarrow \frac{|[u]_E \cap A|}{|[u]_E|} \geq p \quad (19)$$

$$u \in \overline{\text{apr}}_E^q(A) \Leftrightarrow \frac{|[u]_E \cap A|}{|[u]_E|} > q \quad (20)$$

where $1 \geq p > q \geq 0$ are parameters of the model. In other words, an object belongs to the VPRS lower approximation if at least a fraction p of its equivalence class belongs to A , while it belongs to the upper approximation if more than a fraction q of $[u]_E$ is inside A . The model assumes that U is finite (which is not a problem considering that its application is in data analysis), and that $p > q$, to ensure that $\underline{\text{apr}}_E^p(A) \subseteq \overline{\text{apr}}_E^q(A)$. When $p = 1$ and $q = 0$, we recover Pawlak's original equations (3) and (4). In general, probabilistic rough set approaches have been exploited successfully for classification purposes, most notably within Yao's framework of three-way decisions [61].

Ziarko's VPRS model served as an inspiration source for different robust fuzzy rough set proposals. One of them, the Vaguely Quantified Rough Set (VQRS) model [5] softens the membership criterion for an object u to belong to the lower approximation of A into "most elements of $[u]_E$ are inside A ". Similarly, u belongs to the VQRS upper approximation of A to the extent that "at least some elements in $[u]_E$ belong to A ". To formalize this idea, the inherently fuzzy quantifiers "most" and "at least some" are modeled as specific fuzzy sets in the unit interval [63]:

Definition 4.1: A fuzzy set Q in $[0, 1]$ is called a regular increasing monotone (RIM) quantifier if Q is non-decreasing, $Q(0) = 0$ and $Q(1) = 1$.

The class of RIM quantifiers includes as special cases the existential and the universal quantifier:

$$Q_\exists(x) = \begin{cases} 0, & x = 0 \\ 1, & x > 0 \end{cases} \quad Q_\forall(x) = \begin{cases} 0, & x < 1 \\ 1, & x = 1 \end{cases}$$

Examples of RIM quantifiers that also take on values from the interior of the unit interval can be obtained using the following parametrized formula [5], for $0 \leq \alpha < \beta \leq 1$, and x in $[0, 1]$,

$$Q_{(\alpha, \beta)}(x) = \begin{cases} 0, & x \leq \alpha \\ \frac{2(x-\alpha)^2}{(\beta-\alpha)^2}, & \alpha \leq x \leq \frac{\alpha+\beta}{2} \\ 1 - \frac{2(x-\beta)^2}{(\beta-\alpha)^2}, & \frac{\alpha+\beta}{2} \leq x \leq \beta \\ 1, & \beta \leq x \end{cases}$$

For example, $Q_{(0.1, 0.6)}$ and $Q_{(0.2, 1)}$ could be used to represent the fuzzy quantifiers "at least some" and "most" from natural

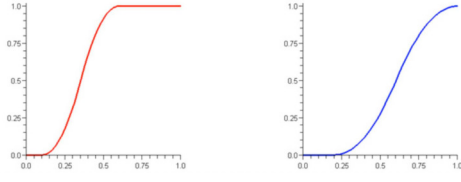


Fig. 1. The RIM quantifiers $Q_{(0.1,0.6)}$ (left) and $Q_{(0.2,1)}$ (right)

language. They are depicted in Figure 1. In general, assuming RIM quantifiers Q_1 and Q_2 such that $Q_1 \subseteq Q_2$, we may define the VQRS approximations of A : for $u \in U$,

$$\underline{\text{apr}}_E^{Q_1}(A)(u) = Q_1\left(\frac{|[u]_E \cap A|}{|[u]_E|}\right) \quad (21)$$

$$\overline{\text{apr}}_E^{Q_2}(A)(u) = Q_2\left(\frac{|[u]_E \cap A|}{|[u]_E|}\right) \quad (22)$$

The above definition has the peculiarity that although both the set to be approximated and the equivalence relation are non-fuzzy, the resulting approximations may well be fuzzy. The reasoning behind this is that for example the membership degree in Eq. (21) evaluates the degree of fulfilment of the condition “ Q_1 elements of $[u]_E$ are in A ”. Note that if $Q_1 = Q_\forall$ and $Q_2 = Q_\exists$, we again arrive at Pawlak’s lower and upper approximation, while the VPRS equations (19) and (20) are also special cases of (19) and (20) using the RIM quantifiers

$$Q_1(x) = \begin{cases} 0, & x \leq p \\ 1, & x \geq p \end{cases} \quad Q_2(x) = \begin{cases} 0, & x < q \\ 1, & x > q \end{cases}$$

The VQRS equations may be further generalized to a fuzzy set A and a fuzzy relation R ; for details, we refer to [5]. Despite its intuitive appeal, the VQRS model has an important shortcoming which it shares with the VPRS model: it does not satisfy relation monotonicity, Eq. (6). This is in particular problematic in applications where the (fuzzy) indiscernibility relation is iteratively refined by adding more information (additional attributes). For example, in [7] it is shown how this affects the operation of the greedy QuickReduct feature selection algorithm.

A solution to this problem can be found by revisiting the equations (15) and (16) and replacing the inf and sup operators by less extreme ones. First note that for finite universes, inf and sup correspond to min and max, respectively. So, the lower approximation (15) is determined solely by the smallest one among all $I(R(v, u), A(u))$ values, and the single largest value $T(R(v, u), A(u))$ will set the upper approximation. A more balanced evaluation is offered by using ordered weighted average (OWA) operators [57]: given an input vector of $n \geq 1$ real values $\langle a_1, \dots, a_n \rangle$ and a weight vector $W = \langle w_1, \dots, w_n \rangle$ such that each $w_i \in [0, 1]$ and $\sum_{i=1}^n w_i = 1$, we first order the input values a_i from large to small obtaining $\langle c_1, \dots, c_n \rangle$ and then compute

$$OWA_W \langle a_1, \dots, a_n \rangle = \sum_{i=1}^n w_i c_i \quad (23)$$

This leads to the definitions of the OWA-based lower and upper approximation [9]:

$$\underline{\text{apr}}_R^{W_1}(A)(u) = OWA_{W_1} \underbrace{I(R(v, u), A(u))}_{v \in U} \quad (24)$$

$$\overline{\text{apr}}_R^{W_2}(A)(u) = OWA_{W_2} \underbrace{T(R(v, u), A(u))}_{v \in U} \quad (25)$$

Using $W_1 = \langle 0, \dots, 0, 1 \rangle$ and $W_2 = \langle 1, 0, \dots, 0 \rangle$, we obtain the original Eq. (15) and (16). Because of the monotonicity properties of T , I and the OWA operator, relation monotonicity is guaranteed, and moreover it always holds that

$$\underline{\text{apr}}_R(A) \subseteq \underline{\text{apr}}_R^{W_1}(A) \text{ and } \overline{\text{apr}}_R^{W_2}(A) \subseteq \overline{\text{apr}}_R(A) \quad (26)$$

In other words, OWA-based fuzzy rough sets indeed relax the original definitions, enlarging the lower approximation and restricting the upper one. In [56], different weighting schemes were discussed and evaluated experimentally.

Recently, in [41] it was shown that for certain choices of the t-norm T (including the product and Łukasiewicz t-norm, but not minimum), the OWA-based lower and upper approximations of any fuzzy set A are exact sets, i.e.

$$\underline{\text{apr}}_R(\underline{\text{apr}}_R^{W_1}(A)) = \overline{\text{apr}}_R(\underline{\text{apr}}_R^{W_1}(A)) = \underline{\text{apr}}_R^{W_1}(A) \quad (27)$$

$$\underline{\text{apr}}_R(\overline{\text{apr}}_R^{W_2}(A)) = \overline{\text{apr}}_R(\overline{\text{apr}}_R^{W_2}(A)) = \overline{\text{apr}}_R^{W_2}(A) \quad (28)$$

This means in particular that these approximations possess a granular representation, and can be used as a basis for fuzzy rule induction algorithms, as discussed in the next section.

V. DISCUSSION: CHALLENGES AND OPPORTUNITIES FOR FUZZY ROUGH SETS

Even though the VQRS model, considering its violation of the relation monotonicity property, has been mostly abandoned in favour of the OWA-based approach, its interpretation of membership to the fuzzy-rough approximations in terms of fuzzy quantifiers expressing “most” and “at least some” is arguably more intuitive and transparent than the one using the rather less compact representation of OWA weight vectors.

Yet this does not mean that OWA fuzzy rough sets are isolated from vague quantification. In fact, as explained in [49], from any OWA weight vector $W = \langle w_1, \dots, w_n \rangle$, a corresponding RIM quantifier Q can be derived by setting

$$Q(x) = \sum_{i \leq xn} w_i \quad (29)$$

and, vice versa, for every RIM quantifier Q and $n \geq 1$, the associated OWA weights w_i ($i = 1, \dots, n$) are determined by

$$w_i = Q\left(\frac{i}{n}\right) - Q\left(\frac{i-1}{n}\right) \quad (30)$$

As such, the weight vectors W_1 and W_2 featured in Eq. (24) and (25) are mutually interchangeable with their VQRS counterparts Q_1 and Q_2 . The resulting membership degrees, however, carry a different meaning; for example, $\underline{\text{apr}}_R^{W_1}(A)(u)$ should be understood as the truth value of the statement “for

most objects in U , it holds that if they are indiscernible from u , then they also belong to A "; in other words, the quantification also takes into account objects completely unrelated to u (i.e., fully discernible from u), while for the computation of the membership to the VQRS lower approximation, these objects are excluded.

A more serious limitation of the OWA fuzzy rough set model lies in the fact that it treats all objects symmetrically during the aggregation process, i.e., an individual object's impact is determined merely by its fulfillment of a specific logical formula. In practice, this means that we hypothesize that "a limited amount" of objects are outliers, and that their effect will be cancelled out by the chosen weighting scheme. Suppose however that we have specific knowledge that some specific objects are in fact certainly outliers, e.g., based on an outlier score that was calculated for them separately. Then, a more natural way to evaluate whether an object u belongs to the lower approximation of concept A is by checking if all objects indiscernible from u , *except* perhaps those which are considered outliers, belong to A .

In order to accommodate the above and other related use case scenarios, new definitions were proposed for the fuzzy-rough lower and upper approximation in [49] recently. They are based on the Choquet integral, a generalization of the classical Lebesgue integral to non-additive measures which has become popular as an aggregation function in decision making [20]. It was shown how the resulting Choquet-based fuzzy rough sets (CFRS) contain the OWA-based model as a special case, while inheriting some of its desirable properties, including set and relation monotonicity. At the same time, they also maintain the intuitive interpretation in terms of vague quantification. Considering that most of the existing fuzzy rough set models still rely on "traditional" approaches to fuzzy quantifiers such as those proposed by Zadeh [63] and Yager [58], which were shown to suffer from some serious conceptual flaws [19], this opens up exciting research opportunities involving more recent developments (see e.g. [15] for an overview).

An important unresolved question about the new CFRS model is whether it still conforms to the granular structure that the OWA-based model from Eq. (24) and (25), and the traditional fuzzy rough set model from Eq. (15) and (16) exhibit, in other words: whether its approximations are exact fuzzy sets in the sense of Eq. (18). While technical in nature, if this question can be answered positively, it opens the doors to fuzzy rule induction methods based on these fuzzy-rough approximations. Indeed, the fuzzy granules corresponding to the approximations can be used inside the antecedent part of fuzzy rules, and an unseen test object's membership in them may be interpreted as the firing strength of the corresponding rule.

As a concrete example, let us consider rule-based classification. In this case, U is partitioned into a number of decision classes (concepts). Let C be one such decision class, and denote its lower approximation, computed according to one of the models discussed in this paper, by $\underline{apr}(C)$. Then, if

$\underline{apr}(C)$ is granularly representable, by Eq. (18) a corresponding decision rule will be generated for every training object u , such that for a given test object v , the firing strength of this rule is obtained as

$$T(R(v, u), \underline{apr}(C)(u)) \quad (31)$$

in other words, as a conjunction between $R(v, u)$, the observed similarity between u and v , and $\underline{apr}(C)(u)$, the membership of the training object u to the lower approximation.

Decision rules based on the lower approximation are usually called "certain" rules, while we refer to those based on the upper approximation as "possible" rules, distinguishing their relative strength. More generally, fuzzy decision rules can be derived from any granularly representable fuzzy set associated to decision classes, for instance from the so-called granular fuzzy-rough approximation introduced in [42], which is defined as the closest granularly representable fuzzy set (w.r.t. a certain loss function) to a given concept, and which is obtained as the result of a linear programming problem.

In practice, however, generating one rule per training object is not a viable approach, and an important challenge is therefore to design proper rule induction algorithm that can at the same time reduce the number of rules, as well as maximize the number of objects that each rule covers. Such a strategy was already pursued in [28], where it was combined with fuzzy rough set guided feature selection, and various other attempts (see e.g. [22], [38], [65], [66]) have also been made to integrate fuzzy rough sets and rule induction; yet, a convincing proposal of a "fuzzy LEM" classification algorithm is still missing and could represent a breakthrough in this domain, not in the least from the perspective of interpretable machine learning.

Indeed, the generation of compact fuzzy rules benefits the human understanding of classification algorithms based on them, as rule-based models are some of the most interpretable models, and they closely resemble human cognition [1]. In [2], various criteria were distinguished for interpretability at different levels of a fuzzy rule-based system, including linguistic variables and fuzzy granules. An integration of these criteria with fuzzy-rough rule induction is therefore at hand to develop a coherent and compact model of interpretable granular computing. Apart from the granules themselves, an important role should again be reserved for fuzzy quantifiers, as they are useful to summarize knowledge in a concise, linguistic way.

ACKNOWLEDGMENTS

The author would like to thank Marko Palanetić and Adnan Theerens for insightful discussions that helped shape this paper.

REFERENCES

- [1] V. Aleven, Rule-Based Cognitive Modeling for Intelligent Tutoring Systems, Springer, pp. 33–62, 2010.
- [2] J. Alonso, C. Castiello, L. Magdalena, C. Mencar, Explainable fuzzy systems: paving the way from interpretable fuzzy systems to explainable AI Systems, Springer, 2021.
- [3] M. Baczyński, B. Jayaram, Fuzzy implications, Springer, 2008.

- [4] A. Bargiela, W. Pedrycz, The roots of granular computing, in: 2006 IEEE International Conference on Granular Computing, pp. 806–809, 2006.
- [5] C. Cornelis, M. De Cock, A. Radzikowska, Vaguely quantified rough sets, in: Proceedings of 11th International Conference on Rough Sets, Fuzzy Sets, Data Mining and Granular Computing (RSFDGrC2007), Lecture Notes in Artificial Intelligence 4482, 2007, pp. 87–94.
- [6] C. Cornelis, M. De Cock, A.M. Radzikowska, Fuzzy rough sets: from theory into practice, in: Handbook of Granular Computing (W. Pedrycz, A. Skowron, V. Kreinovich, eds.), Wiley, 2008, pp. 533–552.
- [7] C. Cornelis, R. Jensen, A noise-tolerant approach to fuzzy-rough feature selection, in: Proc. 2008 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2008), 2008, pp. 1598–1605.
- [8] C. Cornelis, R. Jensen, G. Hurtado Martín, and D. Ślęzak, Attribute selection with fuzzy decision reducts, *Information Sciences*, **180**(2), 2010, pp. 209–224.
- [9] C. Cornelis, N. Verbiest, and R. Jensen, Ordered weighted average based fuzzy rough sets, in: Proc. 5th International Conference on Rough Sets and Knowledge Technology (RSKT 2010), 2010, pp. 78–85.
- [10] M. De Cock, C. Cornelis, E.E. Kerre, Fuzzy rough sets: beyond the obvious, in: Proc. 2004 IEEE Int. Conf. on Fuzzy Systems, FUZZ-IEEE'04, Volume 1, 2004, pp. 103–108.
- [11] M. De Cock, C. Cornelis, E.E. Kerre, Fuzzy rough sets: the forgotten step, *IEEE Transactions on Fuzzy Systems* **15**(1), 2007, pp. 121–130.
- [12] L. D'eer, N. Verbiest, C. Cornelis, L. Godo, A comprehensive study of implicator-conjunctive based and noise-tolerant fuzzy rough sets: definitions, properties and robustness analysis, *Fuzzy Sets and Systems* **275**, 2015, pp. 1–38.
- [13] C. Degang, Z. Suyun, Local reduction of decision system with fuzzy rough sets, *Fuzzy Sets and Systems* **161**(13), 2010, pp. 1871–1883. Chen Degang, Zhao Suyun,
- [14] C. Degang, Y. Yongping, W. Hui, Granular computing based on fuzzy similarity relations, *Soft Computing* **15**(6), 2011, pp. 1161–1172.
- [15] M. Delgado, M. D. Ruiz, D. Sánchez, M. A. Vila, Fuzzy association rules: general model and applications, *IEEE Transactions on Fuzzy Systems* **11**(2), 2003, pp. 214–225.
- [16] M. Delgado, M. D. Ruiz, D. Sánchez, M. A. Vila, Fuzzy quantification: a state of the art, *Fuzzy Sets and Systems* **242**, 2014, pp. 1–30.
- [17] D. Dubois and H. Prade, Rough fuzzy sets and fuzzy rough sets, *International Journal of General Systems* **17**, 1990, pp. 91–209.
- [18] L. Fariñas del Cerro, H. Prade, Rough sets, twofold fuzzy sets and modal logic—Fuzziness in indiscernibility and partial information, In: A. Di Nola, A.G.S. Ventre, Eds., *The Mathematics of Fuzzy Systems*, Verlag TUV Rheinland, Köln, 1986, pp. 103–120.
- [19] A. Fernández, V. Lopez, M.J. del Jesus, F. Herrera, Revisiting evolutionary fuzzy systems: taxonomy, applications, new trends and challenges, *Knowledge-Based Systems* **80**, 2015, pp. 109–121.
- [20] I. Glöckner, *Fuzzy quantifiers: a computational theory*, *Studies in Fuzziness and Soft Computing* 193, Springer, 2008.
- [21] M. Grabisch, C. Labreuche, A decade of application of the Choquet and Sugeno integrals in multi-criteria decision aid, *Ann. Oper. Res.* **175** (1), 2010, pp. 247–286.
- [22] S. Greco, B. Matarazzo, and R. Słowiński, Rough sets theory for multicriteria decision analysis, *European journal of operational research* **129**(1), 2001, pp. 1–47.
- [23] S. Greco, M. Inuiguchi, R. Slowinski, Fuzzy rough sets and multiple-premise gradual decision rules, *International Journal of Approximate Reasoning* **41**, 2005, 179–211.
- [24] J.W. Grzymala-Busse, LERS—a system for learning from examples based on rough sets, in: *Intelligent decision support*, Springer, 1992, pp. 3–18.
- [25] J.W. Grzymala-Busse, J. Stefanowski, Three discretization methods for rule induction, *International Journal of Intelligent Systems* **16**(1), 2001, pp. 29–38.
- [26] Q. Hu, S. An, X. Yu, D. Yu, Robust fuzzy rough classifiers, *Fuzzy Sets and Systems* **183**(1), 2011, pp. 26–43.
- [27] J. Hühn, E. Hüllermeier, FURIA: an algorithm for unordered fuzzy rule induction, *Data Mining and Knowledge Discovery* **19**(3), 2009, pp. 293–319.
- [28] M. Inuiguchi, W.Z. Wu, C. Cornelis, N. Verbiest, Fuzzy-rough hybridization, in: *Springer Handbook of Computational Intelligence*, 2015, pp. 425–451.
- [29] R. Jensen, C. Cornelis, Q. Shen, Hybrid fuzzy-rough rule induction and feature selection, in: Proc. 2009 IEEE International Conference on Fuzzy Systems, 2009, pp. 1151–1156.
- [30] R. Jensen, C. Cornelis, Fuzzy-rough instance selection, in: Proc. 19th International Conference on Fuzzy Systems (FUZZ-IEEE 2010), 2010, pp. 1776–1782.
- [31] R. Jensen and C. Cornelis, Fuzzy-rough nearest neighbour classification, *Transactions on rough sets*, vol. XIII, 2011, pp. 56–72.
- [32] E.P. Klement, R. Mesiar, E. Pap, *Triangular norms*, Springer, 2000.
- [33] O.U. Lenz, D. Peralta, C. Cornelis, Scalable approximate FRNN-OWA classification, *IEEE Transactions on Fuzzy Systems* **28**(5), 2020, pp. 929–938.
- [34] O.U. Lenz, D. Peralta, C. Cornelis, fuzzy-rough-learn 0.1: A Python library for machine learning with fuzzy rough sets, in: Proc. International Joint Conference on Rough Sets, 2020, pp. 491–499.
- [35] M.J. Lesot, G. Moysse, B. Bouchon-Meunier, Interpretability of fuzzy linguistic summaries, *Fuzzy Sets and Systems* **292**, 2016, pp. 307–317.
- [36] Y. Lin, Y. Li, C. Wang, J. Chen, Attribute reduction for multi-label learning with fuzzy rough set, *Knowledge-based systems* **52**, 2018, pp. 51–61.
- [37] C. Molnar, *Interpretable machine learning*, Lulu.com, 2020.
- [38] G. Nápoles, C. Mosquera, R. Falcon, I. Grau, R. Bello, K. Vanhoof, Fuzzy-Rough Cognitive Networks, *Neural Networks* **97**, 2018, pp. 19–27.
- [39] A. Naumoski, G. Mirceva, K. Mitreski, Novel t-norm for fuzzy-rough rule induction algorithm and its influence, in: *ICT Innovations 2021. Digital Transformation, Communications in Computer and Information Science*, vol. 1521. Springer, 2021, pp. 115–125.
- [40] P. Ni, S. Zhao, X. Wang, H. Chen, C. Li, E.C.C. Tsang, Incremental feature selection based on fuzzy rough sets, *Information Sciences* **536**, 2020, pp. 185–204.
- [41] M. Palangetić, C. Cornelis, S. Greco, R. Słowiński, Fuzzy extensions of the dominance-based rough set approach, *International Journal of Approximate Reasoning* **129**, 2021, pp. 1–19.
- [42] M. Palangetić, C. Cornelis, S. Greco, R. Słowiński, Granular representation of OWA-based fuzzy rough sets, *Fuzzy Sets and Systems*, in press.
- [43] M. Palangetić, C. Cornelis, S. Greco, R. Słowiński, A novel machine learning approach to data inconsistency with respect to a fuzzy relation, arXiv:2111.13447 [cs.AI].
- [44] Z. Pawlak, Rough sets, *International Journal of Computer and Information Sciences* **11**(5), 1982, pp. 341–356.
- [45] Z. Pawlak, A. Skowron, Rudiments of rough sets, *Information Sciences* **177**(1), 2007, pp. 3–27.
- [46] Z. Pawlak, A. Skowron, Rough sets: some extensions. *Information Sciences* **177**(1), 2007, pp. 28–40.
- [47] Z. Pawlak, A. Skowron, Rough sets and boolean reasoning, *Information Sciences* **177**(1), 2007, pp. 41–73.
- [48] E. Ramentol, S. Vluymans, N. Verbiest, Y. Caballero, R. Bello, C. Cornelis, F. Herrera, IFROWANN: imbalanced fuzzy-rough ordered weighted average nearest neighbor classification, *IEEE Transactions on Fuzzy Systems* **23**(5), 2015, pp. 1622–1637.
- [49] A.M. Radzikowska, E.E. Kerre, A comparative study of fuzzy rough sets, *Fuzzy Sets and Systems* **126**, 2002, pp. 137–156.
- [50] A. Theerens, O.U. Lenz, C. Cornelis, Choquet-based fuzzy rough sets, *International Journal of Approximate Reasoning*, in press.
- [51] N. Verbiest, C. Cornelis, F. Herrera, FRPS: a fuzzy rough prototype selection method, *Pattern Recognition* **46**(10), 2013, pp. 2770–2782.
- [52] N. Verbiest, E. Ramentol, C. Cornelis, F. Herrera, Preprocessing Noisy Imbalanced Datasets using SMOTE enhanced with Fuzzy Rough Prototype Selection **22**, 2014, pp. 511–517.
- [53] S. Vluymans, L. D'eer, Y. Saeys, C. Cornelis, Applications of fuzzy rough set theory in machine learning: a survey, *Fundamenta Informaticae* **142**(1-4), 2015, pp. 53–86.
- [54] S. Vluymans, D. Sánchez Tarragó, Y. Saeys, C. Cornelis, F. Herrera, Fuzzy rough classifiers for class imbalanced multi-instance data, *Pattern Recognition* **53**, 2016, pp. 36–45.
- [55] S. Vluymans, A. Fernández, Y. Saeys, C. Cornelis, F. Herrera, Dynamic affinity-based classification of multi-class imbalanced data with one-vs-one decomposition: a fuzzy rough approach, *Knowledge and Information Systems* **6**(1), 2018, pp. 55–84.
- [56] S. Vluymans, C. Cornelis, F. Herrera, Y. Saeys, Multi-label classification using a fuzzy rough neighborhood consensus, *Information Sciences* **433-434**, 2018, pp. 96–114.
- [57] S. Vluymans, N. Mac Parthalaín, C. Cornelis, Y. Saeys, Weight selection strategies for ordered weighted average based fuzzy rough sets, *Information Sciences* **501**, 2019, pp. 155–171.

- [58] R.R. Yager, On ordered weighted averaging aggregation operators in multicriteria decision making, *IEEE Transactions on systems, Man, and Cybernetics* **18**(1), 1988, pp. 183–190.
- [59] R.R. Yager, Quantifier guided aggregation using OWA operators, *International Journal of Intelligent Systems* **11**(1), 1996, pp. 49–73.
- [60] J.T. Yao, A.V. Vasilakos, W. Pedrycz, Granular computing: perspectives and challenges, *IEEE Transactions on Cybernetics* **43**(6), 2013, pp.1977–1989.
- [61] Y. Yao, *Granular computing using neighborhood systems*, Advances in soft computing, Springer, 1999, pp. 539–553.
- [62] Y. Yao, Three-way decisions with probabilistic rough sets, *Information Sciences* **180**(3), 2010, pp. 341–353.
- [63] L.A. Zadeh, Fuzzy sets, *Information and Control* **8**, 1965, pp. 338–353.
- [64] L.A. Zadeh, A computational approach to fuzzy quantifiers in natural languages, in: *Computational linguistics*, Elsevier, 1983, pp. 149–184.
- [65] L.A. Zadeh, Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic, *Fuzzy sets and systems* **90**(2), 1997, pp. 111–127.
- [66] S. Zhao, E.C.C. Tsang, D. Chen, X. Wang, Building a rule-based classifier—a fuzzy-rough set approach, *IEEE Transactions on Knowledge and Data Engineering* **22**(5), 2009, pp. 624–638.
- [67] S. Zhao, Z. Dai, X. Wang, P. Ni, H. Chen, C. Li, An accelerator for rule induction in fuzzy rough theory, *IEEE Transactions on Fuzzy Systems* **29**(12), 2021, pp. 3635–3649.
- [68] W. Ziarko, Variable precision rough set model, *Journal of computer and system sciences* **46**(1), 1993, pp. 39–59.

Individual and Collective Self-Development: Concepts and Challenges

Marco Lippi, Stefano Mariani, Matteo Martinelli, Franco Zambonelli
 Department of Sciences and Methods for Engineering
 University of Modena and Reggio Emilia, Italy
 Email: name.surname@unimore.it

Abstract—The increasing complexity and unpredictability of many ICT scenarios will represent a major challenge for future intelligent systems. The capability to dynamically and autonomously adapt to evolving and novel situations, with a partial or limited knowledge of the domain, both at the level of individual components and at the collective level, will become a crucial need for smart devices acting in many application domains. In this paper, we envision future systems able to self-develop mental models of themselves and of the environment they act in. Key properties will include: learning models of own capabilities; learning how to act purposefully towards the achievement of specific goals; and learning how to act in the presence of others, i.e., at the collective level. In our work, we will introduce the vision of self-development in ICT systems, by framing its key concepts and by illustrating suitable application domains. Then, we overview the many research areas that are contributing or can potentially contribute to the realisation of the vision, and identify some key research challenges.

Index Terms—Self-development, sense of agency, learning, self-adaptation, self-organization.

I. INTRODUCTION

HUMAN infants, since their early months, start experiencing with their own body, moving hands, touching objects, and interacting with people around. Such activities are part of an overall process of self-development (a.k.a. autonomous mental development), which lets them gradually develop cognitive and behavioural capabilities [1]. These skills include the capability to recognize situations around, the sense of self, the sense of agency (i.e., understanding the effect of own actions in an environment), the capability to act purposefully towards a goal, and some primitive social capabilities (i.e., knowing how to act in the presence of others).

The possibility of building ICT systems capable – as humans – of self-developing their own mental and social models and to act purposefully in an environment, is increasingly recognized as a key challenge in many areas of artificial intelligence (AI), such as robotics [2], intelligent IoT and smart environments [3], [4], autonomous vehicles management [5], [6].

Indeed, for small-scale and static scenarios, and for simple goal-oriented tasks, it is possible “hardwire” a model of the environment within a system, alongside some pre-designed plans of action. However, for larger and dynamic scenarios, and for complex tasks, individual components of ICT systems should be able to autonomously (i.e., without human supervision): (i) build environmental models and continuously update them as

situations evolve; (ii) develop the capability of recognizing and modelling the effect of their own actions on the context (which variables of the environment can or cannot be directly affected by which actuators, which variables and actuators relate to each other); and (iii) learn to achieve goals on this basis and depending on the current situation; (iv) learn how to organize and coordinate actions among multiple distributed components whenever necessary.

The main contribution of this paper is to frame the key concepts of self-development in ICT systems and to identify challenges and promising research directions. More in particular: Section II introduces a general conceptual framework for the (continuous and adaptive) process of self-development, both at the individual and at the collective level, and sketches key application scenarios; Section III analyzes the most promising approaches in the area of machine learning, multiagent systems, and collective adaptive systems that can contribute with fundamental building blocks towards realizing the vision of self-development, each *per se* challenging; Section IV identifies additional horizontal challenges to be attacked, emphasizing the key role that the self-adaptive and self-organizing research community could play. Finally, Section V concludes by sketching our current and future research work in the area.

II. FRAMEWORK AND APPLICATIONS

The term “self-development” is used to indicate the process carried out by infants during the early stages of their life [1] but, more generally, it can be also associated to the developmental nature of agents that live and interact with a novel environment. The idea of our framework is depicted in Figure 1. At the *individual* level, the first contacts an agent has with a new environment are through *embodiment and perception*: it typically tries to move and interact, in order to test the effect of its own action, so as to acquire a *sense of agency*. Only after these skills have been sufficiently developed, the agent can start behaving in a *goal-oriented* way, by choosing the sequence of actions that can bring to the fulfilment of a goal.

Clearly, the individual level quickly turns into a *collective* one, where the agent has to face other agents, which are not under its control: thus, the agent learns to recognize *self and non-self*, as well as to develop *strategic thinking*, by choosing its own actions by taking into account the behaviour of the other agents. As the complexity grows, the agent will need to

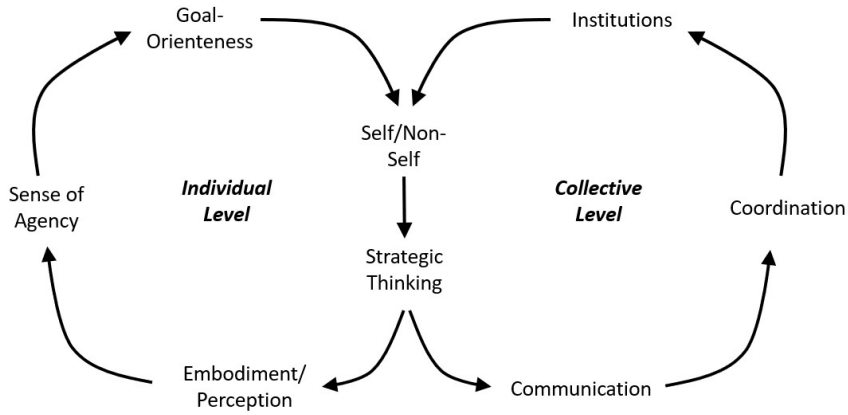


Fig. 1: The conceptual framework of self-development.

understand whether it can *communicate* with others, and with which protocols, as well as *coordinate* in order to jointly act towards a common goal, possibly through the creation of an *institution*.

The whole development, at both the individual and collective level, can be seen as a never-ending, cyclic process, where agents have to continuously adapt to new situations and environments.

A. The Individual Level

At the individual level, an agent \mathcal{X} immersed in an environment can observe (or sense) a set of variables $\mathcal{V} = \{v_1, v_2, \dots, v_n\}$. Internal variables that describe the status of the agent are included in this set as well. The agent can interact with the environment through a set of “actions” $\mathcal{A} = \{a_0, \dots, a_{n-1}, null\}$, including the *null* action.

Embodiment and Perception. Initially, the agent needs to autonomously identify and recognize the components of sets \mathcal{A} and \mathcal{V} : this means that it should get acknowledged of its actuating and sensing skills. Even without resorting to complex AI techniques, methods from the reflective and self-adaptive programming systems can effectively apply in this phase [7] to let the agent dynamically self-inspect its capabilities and start analyzing the observed variables. Still in this phase, the agent can also start acquiring some understanding of the relations between the observed variables over time, as well as some simple prediction capabilities.

Sense of Agency. After the first phases that are mostly dealing with perception, the agent needs to understand what are the effects of each $a_j \in \mathcal{A}$ on \mathcal{V} . This can happen even by chance, with random actions, thus trying to apply actions, without any goal in mind, just to see their effects [2]. Throughout this process, the agent will eventually recognize that, given a current state v^t and the application of an action a_i , the environment will reach (with some probability) a different state v^{t+1} . This mechanism enables the construction of the basic sense of agency [1], and of the sense of causality.

Goal-orientedness. As the agent acquires more sophisticated skills, it can start applying \mathcal{A} with a specific goal in mind. Given the current state v^t and a desired future state v^g (the desired “state of the affairs”), the agent applies the acquired sense of agency by applying the actions that can possibly lead to v^g . This also involves achieving the capability of planning the required sequence of actions to achieve a specific goal.

Self and Non-Self. After an individual agent starts interacting with the environment and testing the effects of its own actions \mathcal{A} , it recognizes that such actions have effect on the environment. As an immediate consequence, it also understands that there are effects that are not under its own control. That is, there are “non-self” entities acting in the environment, too. By learning how to apply \mathcal{A} , the agent also learns the limits of such actions because of non-self entities affecting v^t .

Strategic Thinking. Once the agent has built a world model (how \mathcal{A} affects v^t) and has included the mental models of others (non-self) [8], it can start designing strategies. That is, it can recognise that there are goals that it can possibly (or hopefully) attain only by accounting for the actions of others.

Once again, we remark that self-development is not to be conceived as a “once-and-for-all” process. Rather, it is a life-long process: environments can be dynamic, new variables may become available and thus enable more detailed observations. Also, new actions may become feasible or, the other way round, be no longer be available. This requires the agents to re-tune their learnt sense of agency, and re-think how to achieve goals in isolation and in the presence of non-self entities.

B. The Collective Level

As multiple agents enter the arena, each of them quickly recognizes that there are goals that cannot be achieved in isolation or by simply applying strategic thinking, but they rather need a deep interaction among all the actors. Therefore, as part of their individual self-development, also need to develop some forms of “autonomous social engagement”.

Communication. A first, necessary, step is to identify the way in which agents can communicate and exchange messages. Agents should thus be provided with a specific set of “communication actions”, which could take the form of explicit communication acts (messages) or implicit actions that aim at influencing the others, i.e., by leaving signs in the environment (stigmergy) or by acting in a way that is easily noticeable by others (behavioural implicit communication)[9]. In some cases, the agent has to learn how to receive and send such messages, as a social form of action and perception.

Coordination. When evaluating the possible communication actions, each agent understands the way in which such acts can be exploited to control some environmental variables, and even those that are not (fully) controllable by itself alone. Therefore, such explorative behavior enables the learning of basic forms of coordination, which can be thought of as a social form of learning the sense of agency.

Institution. After exploring coordination protocols, the agents can eventually “institutionalize” their way of interacting. That is, they will learn those acceptable social patterns of coordination, and the set of social norms and social incentives, that enables them to systematically achieve goals together [10].

As in the single-agent setting, a dynamic environment or an evolving agent population may require the above collective process to assume a continuous cyclic nature. We hereby remark that the communication, coordination, and institution stages are not necessary to promote complex goal-oriented collective actions [11]. Yet, whenever communication protocols exist, the self-development process will naturally and gradually learn how to exploit them.

C. Application Scenarios

There are diverse application scenarios that can potentially take advantage of systems capable of self-development.

Robotics is the area which first identified the profitability of building robots capable of self-development [12]. In particular, it is necessary when the robot gets damaged while in operation, and has to develop a novel understand of what it can do according to its residual operational capabilities. At the collective level, the autonomous evolution of communication and coordination capabilities can be of fundamental importance to acquire the capability of the collective to act in unknown and dynamically changing scenarios [13].

Smart factories, as collective robotic systems, can be seen as an aggregated group of components that act together in order to achieve a production goal. Beside their basic scheme of functioning, defined at design time, if one component of the manufacturing system breaks or has some unexpected behaviour, the manufacturing system should ideally adapt to the new situation, and self-develop capabilities of acting so as to overcome the problem without undermining production [14]. The need for adaptability and flexibility is indeed explicitly recognized as a key challenge in Industry 4.0 initiatives [15].

Smart homes can facilitate our interactions with the environment and increase our safety and comfort. We envision

that once a new home is built, its smart devices could start exploring their own individual and collective capabilities, so as to eventually learn how they can affect the home environment, and apply such capabilities once users will start populating it. This will also require to continuously adapt to habits and preferences of users, accommodate new devices and services, tolerate partial failures. Our preliminary experience suggests the feasibility of the vision [3].

Smart cities as well can potentially take advantage of self-development approaches [16]. However, unlike in a smart home, a smart city is not a system free to explore the effect of its actions and interactions, and eventually become capable to act in a goal-oriented way. Thus, for this scenario (but most likely also for smart factories), simulation-based approaches should probably be exploited: system components will be made self-developing in a simulated environment, before being eventually deployed in the real world [6].

III. RESEARCH APPROACHES

The idea of self-development, at both the individual and collective level, has been widely investigated in areas such as cognitive psychology, neuroscience, philosophy, and ethics [2]. We hereby focus on the computational perspective, and in particular on the most recent approaches that can contribute to realise the self-development vision (Figure 2). Although most of these approaches can play a fundamental role and are already providing precious insights on the problems, they still have to attack several challenges to become practical tools for future self-developing systems.

We do not focus here on the basic levels of individual self-development, i.e., perception and embodiment, in that tools already exist to give agents sophisticated sensing abilities (e.g., convolutional neural networks to recognize objects, scenes, and activities [17]) and the capability of controlling their own actuators purposefully.

A. Goal-oriented Learning

The broad area of reinforcement learning shares with our vision the objective of training machines to act in a goal-oriented way in a specific context. However, despite the amazing recent results in the area, in particular with deep Q-learning [18], most current approaches do not aim at building systems with a sense of agency and capable of developing an interpretable world model, but rather at achieving goals based on explicit, domain-based rewards, that are named *extrinsic*. This makes most approaches highly ineffective in scaling up to learning tasks in complex contexts, or across domains, or despite the ever-changing dynamics of the environment.

Curriculum-based approaches to machine learning go somewhat in the direction of gradually developing the capability to act in complex scenarios [19]. The agent is first trained on simple tasks, and the gained knowledge is accumulated and exploited in increasingly complex scenarios, where further skills can thus be effectively learnt. Yet again, most of these approaches do not focus on the development of a world model and of an explicit sense of agency.

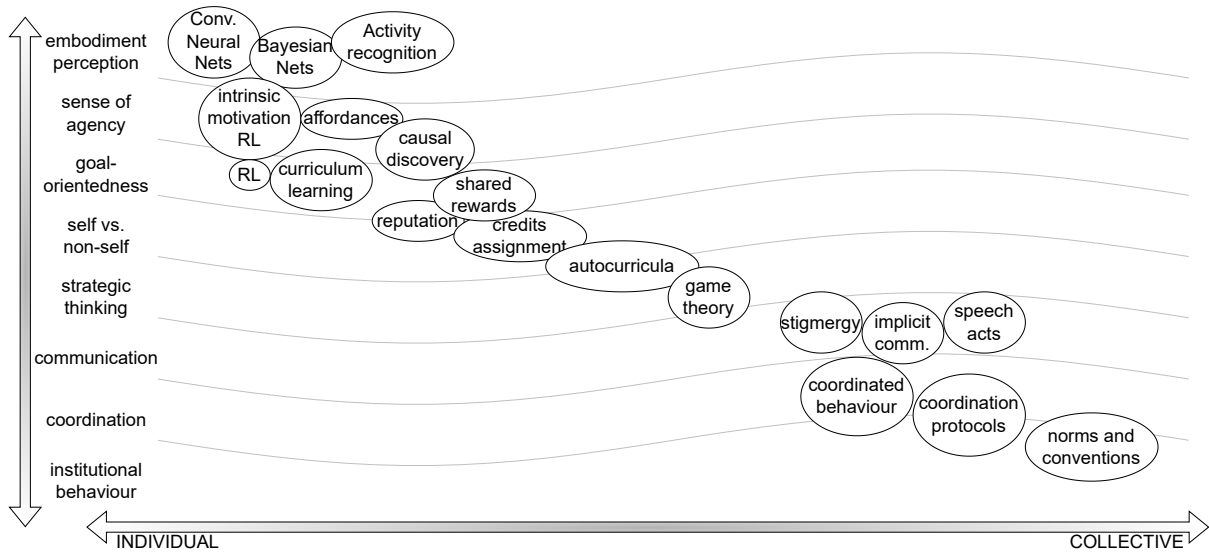


Fig. 2: Key techniques for self-development.

Reinforcement learning approaches based on *intrinsic* rewards [20], instead, more closely exploit the idea of exploring the world to develop a sense of agency. In fact, while extrinsic rewards are typically designed by a “teacher” (e.g., the score in a videogame) intrinsic rewards are developed by the agent itself to satisfy its curiosity (i.e., when it discovers how to achieve specific tasks). For example, in [21] intrinsic rewards are computed as the error in forecasting the consequence of the action performed by the agent given its current state.

Recent approaches based on the theory of affordances [22] propose to have agents gradually learn the effects of their actions. By having them act in constrained environments where only a limited set of actions apply, they eventually develop an explicit sense of agency, i.e., a model of how their actions affect the environment.

In any case, all these approaches face the key challenge of building general conceptual and practical tools to: (i) learn to effectively act in an environment by exploiting the power of model-free sub-symbolic (deep learning) approaches; and, at the same time, (ii) learn incremental and reusable causal models of the world. The latter being increasingly recognized as a key ingredient for intelligence and self-development.

B. Learning causality

Understanding and leveraging *causality* is recognized as a key general challenge for AI in the coming years [23]. Judea Pearl [24] has proposed the idea of a “causal hierarchy” (also named “ladder of causation”) to define different levels of causality recognition and exploitation by an intelligent agent. The first level consists in simply detecting causal relations as associations, whereas the second one assumes the possibility to intervene in the environment and observe the effects of the taken actions. Finally, the third level enables reasoning and planning on the basis of counterfactual analysis. Such layers correspond to some of the phases of the self-development

loop we defined: the first one is mostly involved in the perception phase, whereas the second one is associated to the development of a sense of agency and to recognition of self and non-self. The final layer clearly enables goal-oriented behaviour, strategic thinking, and collective coordination.

Bayesian and causal networks are among the models that are most widely exploited in order to build interpretable causal models of the world [24]. A recent contribution that is in line with the ideas we envision for self-development is the application of curriculum learning to the problem of learning the structure of Bayesian networks [25]. On a pure sub-symbolic level, on the other hand, another recent work proposes to learn causal models in an online setting [26], with the aim to find (and strengthen) causal links between input and output variables.

We argue that key challenges in this area concern, again, understanding how to synergetically exploit symbolic and sub-symbolic approaches to learn, represent, and evolve causal models in self-development scenarios, and how to use them to adaptively achieve goals.

C. Autocurricula

When multiple agents act in a shared environment, their actions and their effectiveness in achieving goals are affected by what others do. Game-theoretic approaches to strategic thinking have deeply investigated this problem and the decision-making processes behind [27]. In this context, it has also been shown that agents can effectively learn in autonomy to improve their performance in dealing with others [28].

However, when moving from theoretical settings (e.g., the prisoner’s dilemma) to complex and realistic scenarios where agents have complex goals (e.g., hide-and-peek in a building), peculiar phenomena arise. The more one agent learns, the more it challenges others, triggering a continuous increase in complexity of behaviour, ultimately enabling to incrementally learn

more sophisticated means to act. This somewhat resembles the increase of complexity that agents face in curricula approaches to reinforcement learning. The key difference being that, in the presence of multiple agents, the increase in complexity and capabilities of agents is promoted and self-sustained by the system itself, hence the term *autocurricula* [11].

Recently, autocurricula-based approaches have produced stunning results in multiagent environments, both cooperative and competitive (e.g., in the hide and seek scenario [29]). And we consider such approaches fundamental towards the self-development of complex agent societies. However, a deep understanding of the process that drives evolution of individual and collective behaviours is still missing, and is a key challenge for the next few years. To this end, providing agents an explicit modelling (possibly in causal terms) of the others' behaviour and of the overall societal behaviour, may be necessary [8]. Also, autocurricula approaches do not currently account for the possibility of interact with other agents, which may indeed be fundamental to improve collective learning.

D. Learning to communicate and coordinate

As already mentioned, agents may communicate and coordinate: by explicit messages, by leaving traces in the environment, or implicitly [9].

These forms of communication are already exploited in multiagent learning, mostly to improve the individual learning process by letting agents share information (e.g., for merging their individual causal models of the world [30]) and coordinate actions. However, these communication approaches are usually assumed as an *innate* capability of agents, rather than one to be learnt. That is, agents have an *a-priori* sense of agency with respect to communication actions, whereas in our self-development vision it should be developed by learning.

For example, with reference to explicit communication acts, [31] proposes a voting game to let agents learn to share a communication language and to develop a strategy to communicate. In [32], it is shown that reinforcement learning can be effectively applied to let agents learn how to communicate in order to achieve a specific effect. In the case of *implicit* communication, instead, forms of implicit behavioural communications have been shown to emerge in simple system components that purposefully move in an environment [33], as they learn to affect others with *ad-hoc* actions. Learning to use stigmergy to effectively coordinate is under-explored in the literature, which instead focuses on the opposite – using stigmergy to boost learning.

In any case, the development of general approaches to let agents develop fully-fledged forms of communication and coordination is still an open challenge, which may call for agents to develop not only a model of the world, but an overall model of the society (i.e., a social sense of agency). as a sort of social sense of agency.

E. Emergence of Institutions

Whereas learning to communicate is about understanding how to use communication to coordinate actions with others,

enabling and sustaining global collective achievement of goals requires “institutionalized” means of acting at the collective level, i.e., a set of shared beliefs and of shared social conventions and norms aimed at ruling collective actions [34]. The mechanisms leading to the spontaneous emergence of institutions in human society, there included the mechanisms to promote and sustain altruistic and cooperative behaviour (e.g., reputation and shared rewards), have been widely investigated [35]. However, most approaches to building multiagent systems assume such mechanisms as *explicitly designed* [34].

Yet, some promising studies related to the emergence of institutionalised behaviour in multiagent systems have been undertaken (see [10] for a recent survey). For instance, [36] proposes a collective learning framework where agents learn to adopt norms in repeated coordination, i.e., agents eventually *learn* that a social norm has emerged, and “institutionalize” their behaviour in their (social) decision making processes by complying to the norm. Another interesting work [37] integrates rational thought, reinforcement learning, and social interactions to model norms emergence in a society: agents incrementally develop a social behaviour (a social norm) while *internalising* it within their cognitive model.

However, the development of general models and tools to support the proper learning and evolution of institutionalised mechanisms of coordination in ICT and multiagent systems is still missing, and so are the solutions to the many problems involved in this process. For instance: how to avoid that an agent learns that free-riding is better than abiding norms; or how to avoid inconsistencies and misunderstandings in their interpretation.

IV. HORIZONTAL CHALLENGES

The presented approaches and techniques are still at the research stage, and many research challenges have been identified for each of them. In addition, it is possible to identify several additional “horizontal” challenges, i.e., of a general nature independently of the specific approach.

The specific nature of such challenges, in our opinion, makes them specifically suited for being pursued by the self-adaptive and self-organising research community, i.e., the ACSOS community at large.

Engineering. Many of the presented approaches are grounded in machine learning, a discipline with plenty of years of research behind, but in which good engineering practice is often neglected, and traditional software engineering problems are sometimes considered mundane. Systems are often developed *ad-hoc* for a specific task or problem domain, with little attention to modularity, reusability, dependability, thus missing the flexibility to adopt them across different domains, tasks, datasets [38]. In addition, given that the diverse approaches presented can each contribute important pieces to the overall vision of self-development, sound engineering approaches are needed to try to integrate such a heterogeneous plethora into a coherent whole. These represent multi-faceted and horizontal research challenges that, in our opinion, could and should be profitably attacked by the self-adaptive and self-organising

research community, due to its inherent software engineering endeavour.

Controlling evolution. Self-development raises the issue of somewhat controlling how behaviours evolve, as individual learns new skills and tasks, and as the collective learns new way of coordinating and acting together. How can we *steer* a learning process towards desired outcomes without putting bias in it? How can we *constrain* the boundaries within which individual and collective behaviours should stay (e.g., in terms of safety)? What *interventions* can we make to re-direct an agent or a collective that has taken an unpredictable or unsafe self-development path? Experience in self-adaptive components based on feedback, as well as in the study of emergent behaviours in self-organising systems and definitely help in finding proper technical answers, and – why not – *ethical* ones [39].

Humans in the Loop. The more self-development technologies will advance, the more humans will have to actively interact with them. This interaction will raise technical issues (will we have “handles” to control or block such systems in some ways and to some extent?) and ethical problems (will we be rather “handled” by these systems and subjects to their decisions?). Some of these problems already emerged, like in the *moral machine* experiment [40] or in AI-based hiring technology. Technical challenges will be meat for the HCI and distributed systems communities (there included the self-organising systems one). Ethical and moral ones will be meat for politicians and lawyers, although deep joint work with technical experts will always be necessary. A key ingredient involves institutions, since they represent humans as a group: laws and regulations need to be developed to regulate the global actors into the day-by-day technology usage. Nevertheless, a deeper interaction between researchers in science and technology and public institutions is needed to support the regulation design phase.

Sustainability. Algorithms for self-development will most likely require extensive computational resources. For example, the mentioned “hide and seek” experiment by OpenAI involved a distributed infrastructure of 128,000 pre-emptible CPU cores and 256 GPUs on GCP [29]: the default model optimised over 1.6 million parameters taking 34 hours to reach the fourth stage over six of agents skills progression. This example is a sort of best-in-class projects; anyway, it is clear that if self-developing systems will be based on similar learning approaches, they will require massive amounts of computational resources. Therefore, a key challenge for the community will be to devise algorithmic and system-level means to make self-development systems sustainable, and affordable by others other than the big technology players.

Explainability. Being able to inspect and explain the decision making process of AI systems is already a hot topic, so much that an entire research field (XAI, from eXplainable AI) has born. We already commented several times how such problems should be compulsory accounted for also for self-development, possibly with the help of causal models. This is indeed a key challenge for self-adaptive and, especially,

for self-organising research, too, where explaining global behaviours, patterns, and configurations emerging from local interactions is mostly still considered the “holy grail”.

V. CONCLUSIONS, CURRENT AND FUTURE WORK

In this paper, we have elaborated upon the vision of self-development, at both the individual and collective level. Although the road towards fully-realizing the vision is still a long one, several ideas in the areas of learning, causality, multiagent systems, are already showing its potential feasibility.

From our side, we are currently experimenting with Bayesian networks and causal models to learn dependencies between variables that represent sensors and actuators within a smart environment. In a simplified smart home setting, we showed how an agent is able to learn the effect of one of its own actions, thus acquiring the sense of agency, the necessary precondition towards goal-orientedness [3]. The training set consists of a collection of observations where the agent performs random actions and observes their effect on the rest of the environment. Once the learning phase is completed, the agent is eventually able to understand what to do to reach the desired state of affairs. At the collective level, our preliminary experiments show how different agents are able to learn to cooperate to achieve a goal they could not achieve individually. We assumed that the agents can share their observations, thus providing training examples to a single data set that can be used to learn a single, general model. By learning from the joint set of observations and actions, the two agents learn that they need to cooperate and to coordinate their actions.

As a continuation of this strand of research, we are now moving to a distributed learning setting, where agents do not fully share their observations to agree on a single global (causal) model of their shared environment. Rather, they cooperate to refine their own local causal models whenever they recognize partial, missing, or wrong information, by organising a coordinated distributed intervention protocol meant to obtain the additional information needed to disambiguate, refine, complete, or correct their own local models.

As part of our future work we plan to investigate how digital twins could enable the learning paradigms described so far. In particular, in many application domains such as smart factories, one could envisage a hierarchical architecture where digital twins collect and integrate data coming from heterogeneous physical devices, building more and more abstract models and representations.

Acknowledgements: Work supported by the Italian MUR, PRIN 2017 Project “Fluidware”.

REFERENCES

- [1] P. Rochat, “Self-perception and action in infancy,” *Experimental brain research*, vol. 123, no. 1-2, pp. 102–109, 1998.
- [2] J. Weng, J. McClelland, A. Pentland, O. Sporns, I. Stockman, M. Sur, and E. Thelen, “Autonomous mental development by robots and animals,” *Science*, vol. 291, no. 5504, pp. 599–600, 2001.

- [3] M. Lippi, S. Mariani, and F. Zambonelli, "Developing a sense of agency in IoT systems: Preliminary experiments in a smart home scenario," in *17th CoMoRea workshop at PerCom*. IEEE, 2021.
- [4] S. Jha, M. Schiemer, F. Zambonelli, and J. Ye, "Continual learning in sensor-based human activity recognition: An empirical benchmark analysis," *Inf. Sci.*, vol. 575, pp. 1–21, 2021.
- [5] S. Mariani, G. Cabri, and F. Zambonelli, "Coordination of autonomous vehicles: Taxonomy and survey," *ACM Comput. Surv.*, vol. 54, no. 1, pp. 19:1–19:33, 2021.
- [6] Y. Yang, M. Taylor, J. Luo, Y. Wen, O. Slumbers, D. Graves, H. Bou Ammar, and J. Wang, "Diverse auto-curriculum is critical for successful real-world multiagent learning systems," in *20th International Conference on Autonomous Agents and Multiagent Systems*. IFAAMAS, 2021.
- [7] M. Salehie and L. Tahvildari, "Self-adaptive software: Landscape and research challenges," *ACM transactions on autonomous and adaptive systems*, vol. 4, no. 2, pp. 1–42, 2009.
- [8] B. Subagdja and A.-H. Tan, "Beyond autonomy: The self and life of social agents," in *Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems*, 2019, pp. 1654–1658.
- [9] S. Mariani and A. Omicini, "Anticipatory coordination in socio-technical knowledge-intensive environments: Behavioural implicit communication in MoK," in *AI*IA 2015*. Springer, 2015, pp. 102–115.
- [10] A. Morris-Martin, M. De Vos, and J. Padget, "Norm emergence in multiagent systems: a viewpoint paper," *Autonomous Agents and Multi-Agent Systems*, vol. 33, no. 6, pp. 706–749, 2019.
- [11] J. Z. Leibo, E. Hughes, M. Lanctot, and T. Graepel, "Autocurricula and the emergence of innovation from social interaction: A manifesto for multi-agent intelligence research," *arXiv preprint arXiv:1903.00742*, 2019.
- [12] J. Bongard, V. Zykov, and H. Lipson, "Resilient machines through continuous self-modeling," *Science*, vol. 314, no. 5802, pp. 1118–1121, 2006.
- [13] N. Cambier, R. Miletitch, V. Frémont, M. Dorigo, E. Ferrante, and V. Trianni, "Language evolution in swarm robotics: A perspective," *Frontiers in Robotics and AI*, vol. 7, p. 12, 2020.
- [14] M. Martinelli, S. Mariani, M. Lippi, and F. Zambonelli, "Self-development and causality in intelligent environments," in *Workshops at 18th International Conference on Intelligent Environments (IE2022), Biarritz, France, 20-23 June 2022*, ser. Ambient Intelligence and Smart Environments, vol. 31. IOS Press, 2022, pp. 248–257.
- [15] J. Zhang, X. Yao, J. Zhou, J. Jiang, and X. Chen, "Self-organizing manufacturing: Current status and prospect for industry 4.0," in *5th International Conference on Enterprise Systems*, 2017, pp. 319–326.
- [16] M. Saelens, Y. Kinoo, and D. Weyns, "Heyciti: Healthy cycling in a city using self-adaptive internet-of-things," in *IEEE International Conference on Autonomic Computing and Self-Organizing Systems Companion*, 2020, pp. 226–227.
- [17] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [18] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.
- [19] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Proceedings of the 26th International Conference on Machine Learning*, 2009, pp. 41–48.
- [20] J. Schmidhuber, "Formal theory of creativity, fun, and intrinsic motivation (1990–2010)," *IEEE Transactions on Autonomous Mental Development*, vol. 2, no. 3, pp. 230–247, 2010.
- [21] Y. Burda, H. Edwards, D. Pathak, A. Storkey, T. Darrell, and A. A. Efros, "Large-scale study of curiosity-driven learning," in *International Conference on Learning Representations*, 2018.
- [22] K. Khetarpal, Z. Ahmed, G. Comanici, D. Abel, and D. Precup, "What can i do here? a theory of affordances in reinforcement learning," in *International Conference on Machine Learning*, 2020, pp. 5243–5253.
- [23] B. Schölkopf, F. Locatello, S. Bauer, N. R. Ke, N. Kalchbrenner, A. Goyal, and Y. Bengio, "Toward causal representation learning," *Proceedings of the IEEE*, 2021.
- [24] J. Pearl and D. Mackenzie, *The book of why: the new science of cause and effect*. Basic Books, 2018.
- [25] Y. Zhao, Y. Chen, K. Tu, and J. Tian, "Learning bayesian network structures under incremental construction curricula," *Neurocomputing*, vol. 258, pp. 30–40, 2017.
- [26] K. Javed, M. White, and Y. Bengio, "Learning causal models online," *arXiv preprint arXiv:2006.07461*, 2020.
- [27] R. B. Myerson, *Game theory*. Harvard university press, 2013.
- [28] A. Nowé, P. Vrancx, and Y.-M. De Hauwere, "Game theory and multi-agent reinforcement learning," in *Reinforcement Learning*. Springer, 2012, pp. 441–470.
- [29] B. Baker, I. Kanitscheider, T. Markov, Y. Wu, G. Powell, B. McGrew, and I. Mordatch, "Emergent tool use from multi-agent autocurricula," *arXiv preprint arXiv:1909.07528*, 2020.
- [30] S. Meganck, S. Maes, B. Manderick, and P. Leray, "Distributed learning of multi-agent causal models," in *IEEE/WIC/ACM International Conference on Intelligent Agent Technology*. IEEE, 2005, pp. 285–288.
- [31] S. Gupta and A. Dukkkipati, "Winning an election: On emergent strategic communication in multi-agent networks," in *International Conference on Autonomous Agents and Multiagent Systems*, 2020, pp. 1861–1863.
- [32] J. N. Foerster, Y. M. Assael, N. De Freitas, and S. Whiteson, "Learning to communicate with deep multi-agent reinforcement learning," *arXiv preprint arXiv:1605.06676*, 2016.
- [33] N. A. Grupen, D. D. Lee, and B. Selman, "Low-bandwidth communication emerges naturally in multi-agent learning systems," *arXiv preprint arXiv:2011.14890*, 2020.
- [34] M. Esteva, J.-A. Rodriguez-Aguilar, C. Sierra, P. Garcia, and J. L. Arcos, "On the formal specification of electronic institutions," in *Agent mediated electronic commerce*. Springer, 2001, pp. 126–147.
- [35] M. A. Nowak, "Five rules for the evolution of cooperation," *Science*, vol. 314, no. 5805, pp. 1560–1563, 2006.
- [36] C. Yu, M. Zhang, and F. Ren, "Collective learning for the emergence of social norms in networked multiagent systems," *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2342–2355, 2014.
- [37] R. Beheshti, "Normative agents for real-world scenarios," in *International Conference on Autonomous Agents and Multi-Agent Systems*, 2014, pp. 1749–1750.
- [38] B. Porter and R. Rodrigues Filho, "Distributed emergent software: Assembling, perceiving and learning systems at scale," in *IEEE International Conference on Self-Adaptive and Self-Organizing Systems*, 2019, pp. 127–136.
- [39] A. Yapo and J. Weiss, "Ethical implications of bias in machine learning," in *51st Hawaii International Conference on System Sciences*, 2018.
- [40] E. Awad, S. Dsouza, R. Kim, J. Schulz, J. Henrich, A. Shariff, J.-F. Bonnefon, and I. Rahwan, "The moral machine experiment," *Nature*, vol. 563, no. 7729, pp. 59–64, 2018.

Rough Sets Turn 40: From Information Systems to Intelligent Systems

Andrzej Skowron

Systems Research Institute

Polish Academy of Sciences, Warsaw, Poland

Cardinal Stefan Wyszyński University, Warsaw, Poland

Email: andrzej.skowron@gmail.com

Dominik Ślęzak

Institute of Informatics

University of Warsaw, Warsaw, Poland

QED Software Sp. z o.o., Warsaw, Poland

Email: slszak@mimuw.edu.pl

Abstract—The theory of rough sets was founded by Zdzisław Pawlak as a framework for data and knowledge exploration. His seminal paper titled “Rough Sets” was published in 1982, in *International Journal of Computer and Information Sciences*. One of the key aspects that lets us use rough sets in practical scenarios is the notion of information system, which comes from even earlier Professor Pawlak’s works. Information systems are the means for data and knowledge representation. They constitute the input to rough set mechanisms aimed at computing approximations of concepts and deriving compacted, interpretable decision models. In particular, the fundamental notion of the indiscernibility relation is defined on the basis of a given information system. Accordingly, we discuss to what extent information systems can serve as the basis for intelligent systems. We claim that in many cases it is not enough to treat a data set – represented as an information system – as a purely abstract object with no linkage to the data origins. Oppositely, we should give ourselves a technical possibility to construct information systems dynamically, taking into account interaction with physical environments where the data comes from. With this respect, we refer to the notions of interactive granular computing and we generally consider together the paradigms of rough sets, information systems, and information granulation.

Index Terms—Rough Sets; Information Systems; Data Mining; Big Data; Interactive Granular Computing; Intelligent Systems

I. INTRODUCTION AND BASIC CONCEPTS

ZDZISŁAW Pawlak (1926-2006) founded the theory of rough sets with the aim of analyzing incomplete information by means of approximations [1]. His book published in 1991 [2] established worldwide recognition of rough sets as an approach which can model complex problems using simple constructs. The idea of rough sets originated from earlier Professor Pawlak’s research on knowledge representation and information retrieval [3], [4]. The key observation was that we often operate with objects (cases, instances) which are indistinguishable from each other, so approximating the sets of objects by using indistinguishable “blocks” is the most reasonable thing we can do. A need for reaching such simple solutions was important for Professor Pawlak during his entire scientific career which included far more achievements than just rough sets [5]. It is also interesting to note that the basic models he used working in different fields, such as conflict analysis [6] or concurrency [7], were based on information systems. In the same time, his personal interests were going far beyond computer science and science in general [8].

The updated (after 25 years) viewpoint on rough sets was presented in [9], [10], [11]. As it happens with every theory, it took time to understand the actual contribution of rough sets with respect to other approaches. Firstly, rough sets were compared with the theory of fuzzy sets [12]. Over the years, it turned out that these two theories can be successfully combined because they offer complementary granules understood as computational building blocks for approximations [13], [14]. In the next sections, one can see a number of examples of such combinations. Another thread of comparisons was devoted to the principles of information granulation and granular computing (GrC) [15], [16], whereby it is generally assumed to operate on groups of objects / instances / items gathered together into granules. In this case, the relationship turned out to be even more natural because such granules (or generally, various frameworks and layers of granular information) are a natural input for calculations related to rough sets.

Nowadays, rough sets are particularly popular in the area of learning decision models from the data. There are many rough set methods aiming at feature selection and interpretable decision model construction [17], [18], [19]. However, the principles of rough sets have much broader influence. They have an impact on various methodologies of decision making, *e.g.* multi-criteria decision making [20] or three-way decision making which strongly relies on rough set positive, negative and boundary regions [21]. Moreover, rough sets are employed to enhance expressive and algorithmic capabilities of many approaches to data mining and knowledge discovery. A good example here is an extension of standard data clustering toward rough clustering (whereby we search and operate with lower and upper approximations of rough data clusters) and its fuzzy hybridizations [22], [23], [24]. Another example corresponds to formal concept analysis [25] which has this year its 40th anniversary exactly like rough sets. With that respect, there are many approaches to constituting rough set approximations of formal concepts or in other words, formal concepts which comprise of rough set approximations [26]. Rough sets turned out to be useful also in other fields of science and industry. For instance, they were adopted in design of database engines and solutions, with respect to query language extensions (which can be referred as rough querying) and acceleration of commercial database software performance [27], [28].

In all above scenarios, rough set methods (or mechanisms that adopt rough set principles) require an input to derive approximations of concepts, relations, etc. In [1], Professor Pawlak discussed several examples of domains, whereby the inputs to rough set methods could take different forms. One of those forms corresponded to the notion of an information system [3]. In the framework proposed by Professor Pawlak, information systems are aimed at representing the underlying data, information or knowledge that we want to use to describe (and approximate) the concepts of practical interest. An information system comprises of objects, attributes and the values of attributes over those objects. Information systems resemble data tables in the theory of relational databases. It should be noted that data tables, in particular decision tables, have been studied and used in many applications since 60-ties of the XX-century [29], [30]. Still, we need to remember that Professor Pawlak's goal to operate with information systems was to represent information (including information derived from the data) rather than focus on the data itself [2]. This topic is actually wider and one can find analogies between information systems and data / information / knowledge representation frameworks in some other theories [25], [31], [32].

The complete idea of information systems has been created thanks to cooperation of Professor Pawlak with other scientists [4], [33]. At that time, there was a great demand for constituting the foundations for information representation and retrieval [34]. All those works were reaching beyond a standard understanding of data storage and processing, particularly with respect to incompleteness, imprecision and indeterminism of information that one needs to handle in practice. Over the years, the concept of non-deterministic information systems evolved in many interesting directions [35], [36]. Going back to the principles of information granulation [16], one may say that such information systems comprise of descriptions – also called signatures – of granules (composed in different ways, specific to different applications) in terms of available attributes, where those signatures are not always precise.

Operating with granules introduces a useful abstraction, a kind of border between granules' signatures (being the inputs to further computations) and granules' internals, whereby one can locate guidelines about *how* and / or *what for* those signatures are derived from physical data sources. This is why in this paper, we are interested equally in: (i) rough sets as the methodology for deriving concept approximations and compacted decision models, (ii) information systems which provide the input to those derivations, and (iii) the paradigms of GrC and information granulation, as various forms of granules can be "hidden right under" the abstract descriptions that information systems consist of. We focus particularly on the methodology of interactive granular computing (IGrC) [37] and interactive information systems [38], where the above-mentioned abstraction was explicitly introduced in terms of complex granules (c-granules) with the embedded control mechanisms that decide how their signatures are derived.

IGrC raised from the observation that traditional ways of computing do not take into account how the process of

perceiving attribute values is realized, *where* and *when* to access the concerned objects in a physical space, and *why* particular attributes are selected. This kind of awareness / attention / agency [39] is important for designing intelligent systems which are supposed to deal with complex phenomena in the real physical world [40]. This becomes even more important when one realizes that unexperienced data scientists often treat the available data sets as an ultimate baseline without investigating how those sets were collected.

In the above considerations, we evolved from information systems toward intelligent systems. We referred to the concepts of awareness and perception (perceiving attribute values), we can also refer to the concept of cognition. With this respect, let us cite the following statement of Leslie Valiant¹, which is particularly relevant when extending GrC toward IGrC:

A fundamental question for artificial intelligence is to characterize the computational building blocks that are necessary for cognition.

The above computational building blocks can be treated as a generalization of granules known from GrC [15]. (In particular, indistinguishable blocks / indiscernibility classes known from rough sets can be treated as atomic granules.) Naturally, such granules / blocks / classes have been already studied within GrC in the context of cognition [14]. However, when moving from blocks to computational building blocks, it is indeed useful to rely on IGrC because therein, the aforementioned c-granules are aimed at more tasks than just storing their signatures. Such c-granules build the relevant configurations of physical objects, initiate and modify interactions between them, so they are generally responsible for perceiving the physical world. They link physical objects with abstract objects used to represent the instances of decision making from the viewpoint of models working on information systems.

In the next sections, we consider some examples of challenges with respect to which IGrC can be worth adapting. For now, let us mention just one of them, namely hierarchical learning [41], [42]. From a logical viewpoint, one can think about it as learning satisfiability relations at different levels of hierarchy [43], [44]. This includes learning logical structures (e.g. relational structures or models), as well as logical formulas and their semantics expressed using those structures. The current methods of hierarchical learning are often based on GrC, with a special emphasis on designing hierarchies of information systems by basing on domain knowledge [45], [46]. However, granules on which the corresponding reasoning pipelines are performed, cannot neglect the underlying hierarchies of physical objects that are crucial for perception processes. Also, different layers of hierarchy can be connected to different types of sensors and actuators. Thus, the IGrC framework can be helpful indeed to embrace both, the relationships between information systems at different levels of hierarchy and the relationships between particular systems and the associated physical-object-related information sources.

¹people.seas.harvard.edu/~valiant/researchinterests.htm

In the rest of the paper, in Section II, we refer to some selected literature on rough sets. This section is quite extensive given the anniversary flavor of the paper. In Section III, we go back to the discussion about the importance of information systems. We emphasize a need of operating with information systems (and the results of rough set computations over information systems) considered in a wider context of interactions between abstract and physical objects of different sorts. We go through several aspects of applications, whereby this kind of interaction is needed. We show to what extent the paradigms of IGrC can be helpful. We also refer to some concepts known from the domain of big data in order to put our discussion into other contexts. In Section IV, we conclude the paper.

Let us reemphasize the retrospective context of this paper. Besides the 40th anniversary of rough sets (which is our major focus) and the 40th anniversary of formal concept analysis (which was mentioned above), there are two more celebrations worth mentioning. The first of them refers to the rough set workshop series which “visited” the FedCSIS conferences for the first time 10 years ago² and which is now back to the program of technical sessions³. Secondly, this year’s FedCSIS hosts the 30th International Symposium on Concurrency, Specification and Programming (CS&P 2022)⁴. The topics related to rough sets and information systems have been always visible at the CS&P events. In particular, the above-cited papers [36], [40], [44] come from CS&P.

II. SELECTED RELATED WORK ON ROUGH SETS

In order to provide a better viewpoint of the theory and applications of rough sets, we refer to two events from the past. These references will also constitute a better background for our major goal in this paper, which is the review of new advances on rough-set-related information systems.

A. Rough Sets at FedCSIS 2012

The first considered event is FedCSIS 2012 held 10 years ago in Wrocław, Poland⁵. As already mentioned, that was the first time when rough sets occurred so intensively at a FedCSIS conference. Let us start outlining the FedCSIS 2012 rough-set-related publications from [47], [48]. The first paper equips the Variable Precision Rough Set (VPRS) approach [49] with a Bayesian background [50]. The second paper combines VPRS with fuzzy rough set methods [24] in order to produce flexible decision rules. In summary, both papers deal with information imprecision – modeled by probabilities (which is the domain of VPRS) and fuzziness (which can be used to work *e.g.* with partial matching of rules’ antecedents) – and attempt to extract interpretable decision models from the data [11].

The topic of rough-set-driven decision rules is considered also in one more FedCSIS 2012 publication [51]. In general, one will see throughout our whole paper that rough set principles fit the field of rule induction very well [18], [31].

This relationship is evident not only at a technical algorithmic level but also with respect to the common assumption of looking at the data through the glasses of information granules [16]. For more examples of connections between the worlds of rules and rough sets, let us refer *e.g.* to [17], [52].

Going further, papers [53], [54] introduce new heuristic measures that can be used during attribute reduction. It is worth noting that attribute reduction – or in other words algorithmic elimination of redundant attributes from the constructed set of attributes – is an important contribution of rough set research to knowledge discovery and in particular to its phase of feature selection [55]. As a complement to typical feature selection algorithms which attempt to add the most useful attributes, rough set methods take as input the sets of attributes produced by those typical algorithms and attempt to additionally compact them by eliminating unnecessary or approximately unnecessary elements. The additional aspect of attribute elimination occurs in just a few machine learning methodologies worldwide [56], so we can indeed say that this is an important rough sets’ contribution to this area.

Papers [57], [58] continue the topic of attribute reduction. The first of them proposes greedy algorithms for deriving so-called superreducts from data sets with multivalued decision attributes (target variables). It is one more example of dealing with information imprecision in rough set frameworks. Superreducts are the subsets of attributes which are sufficient to induce values (or as in this case, the sets of possible values) of decision attributes. It is also important to note that the notion of superreduct is equivalent to the notion of test in the test theory [31]. The second paper compares the notions of decision bireduct (aimed at deriving both the sets of attributes and the sets of data objects for which those attributes are sufficient to construct rule-based decision models) and approximate reduct (aimed at eliminating as many attributes as possible, even if the ability to induce decisions is not fully preserved). This comparison was later extended in [17].

The next two papers extend the topic of feature selection toward some of modern data challenges, namely high-dimensionality and large data volumes. Paper [59] combines attribute reduction with attribute clustering. Attributes are first grouped using some rough-set-inspired measures and then the methods of attribute reduction work iteratively on cluster representatives. This allows for decreasing the complexity of attribute reduction for high numbers of attributes and it also improves interpretability of results. These methods were later extended to let them work with attribute groups which can be set up for many reasons, including heterogeneity of data sources that are required to derive attribute values [60].

Paper [61] copes with big data volumes by putting attribute reduction and decision tree induction into a relational database framework, whereby the corresponding algorithms are implemented in SQL. The authors extend some previous ideas in this field [19] and, in particular, employ an open source database engine called Infobright Community Edition to run experiments. Infobright Community Edition is an example of using rough sets to optimize other types of data computations,

²fedcsis.org/2012/rsa.html

³fedcsis.org/2022/rsta

⁴fedcsis.org/2022/csp

⁵fedcsis.org/2012/

in this case – query execution in relational databases⁶. This emphasizes that rough sets can be successfully used not only for machine learning and data mining but also for other tasks of big data processing. We refer to [62] for current developments related to Infobright Community Edition. We also refer to [63], where the Infobright’s technology performance is explained in terms of rough set operations on specifically aggregated (granulated) multivalued information systems.

The last two rough-set-related publications are interesting from the information systems’ viewpoint as well. In [64], the source of building an information system is a transformed ontological graph which encodes our knowledge about a given area [65]. The rules derived using the Dominance Rough Set Approach (DRSA) [20] express useful regularities within the original graph. This is actually an illustration of the fundamental idea behind information systems, namely, that such systems may contain not only the empirical data but they may also integrate it with domain knowledge [11], [43].

Finally, paper [66] presents the real-world application of rough sets to explore medical data. Herein, the information system – the input for rough-set-based model learning methods – does not correspond directly to the original data measurements. It is rather a result of a sequence of time-window-driven data aggregations which are typical for building hierarchical information systems describing complex objects [45], [67]. This work applied in particular a rough-set-based software system for machine learning and data mining – called RSES – which is now available in a library format [68] (see also the RSES extension targeted at spatio-temporal concepts⁷). It is also one more practical use case of deploying the Infobright Community Edition database engine to run the underlying operations over granulated and compressed data sets.

B. Rough Set Contest at PP-RAI 2022

The second considered event is the PP-RAI 2022 conference held this year in Gdynia, Poland⁸. The chairs of PP-RAI 2022 decided to celebrate the 40th anniversary of rough sets by organizing the contest for the most influential article on rough sets co-authored by Polish researchers in 2020 or later⁹. Let us discuss below the articles submitted to this contest.

Papers [69], [70] operate at the edge of rough sets and formal concept analysis [25]. The first paper adopts the principles of attribute reduction (or more generally, model compaction) to simplify so-called fuzzy concept lattices, introduced as the means for representing patterns and regularities hidden in numerical data [71]. The second paper is actually an extension of the previously-cited work [36]. The authors attempt to put classical rough sets, formal concept analysis and the DRSA-style extensions of rough sets [20] into a unified conceptual pipeline aimed at transforming the data – through various forms of (possibly multivalued) information systems [33] – to knowledge. Within such a universal framework, the authors

reconsider special cases of rough set operators known from different approaches. Therefore, one may say that this paper is a direct continuation of the ideas introduced in [1].

Papers [72], [73] link rough sets with logical foundations. The first paper shows how to express reasoning based on the VPRS-style extensions of rough sets [49] within the framework provided by a probabilistic extension of PROLOG [74]. The second paper shows how to reason about the properties of various types of rough set approximations within the framework provided by Mizar – a powerful system for automated proving [75]. Needless to say, such foundations are crucial for every theory, including reasoning within the theory and reasoning about the theory. We refer to [76] for more information about logical background of rough sets.

Papers [77], [78] present further advances in the previously-discussed popular rough set approaches such as the above-mentioned DRSA and fuzzy rough sets, respectively. The first paper uses the statistical learning machinery [79] to give new insights into parameters of probabilistic extensions of DRSA. The second paper, somewhat analogously, attempts to provide new interpretation of fuzzy rough set parameters. This is done by considering a new form of fuzzy granules [80], which consequently leads toward more intuitive derivation of fuzzy rough decision rules. One can say that these two articles fall into the same thematic categories as the previously-considered FedCSIS 2012 publications [47] and [48], respectively.

Papers [81], [82] continue the topic of feature selection. The first paper refers to heuristic attribute evaluation measures and data discretization techniques analogous to those reported in [10], [19]. The second paper seems to be particularly interesting as it extends the already-discussed topic of rough set software packages and libraries [24], [68] toward hardware optimizations that are specific for high performance computing. Such optimizations should be further compared and integrated with other acceleration opportunities, *e.g.* adaptation of MapReduce [60] and analytical database engines [61].

Papers [83], [84] refer to rough set software too. The first paper reports one more package delivering rough set methods for data mining and knowledge discovery. The second paper is about the application of that package to biomedical data mining. This second paper – besides its important experimental results – touches the aspects of visual data analytics [85], [86] and a need of understanding both, the analytical processes and their outcomes by subject matter experts [87], [88].

Papers [89], [90], [91] illustrate more real-world applications of rough set methods in the area of biomedicine. The first paper uses rough set approximations built over neighborhood-based information granules [92]. The remaining two papers confirm the expressive power of the DRSA-based decision rules. They also compare the accuracy of rule-based models with other approaches (such as random forests and logistic regression [93]) and show how to derive the attribute importance (see *e.g.* [94]) from the considered rules.

Papers [95], [96] continue the topic of rule induction. The first paper can be compared to [35], as both of them deal with deriving probabilistic rules from incomplete information sys-

⁶en.wikipedia.org/wiki/Infobright

⁷mimuw.edu.pl/~bazan/roughice/?sLang=en

⁸pp-rai2022.umg.edu.pl/

⁹roughsets.org/newspage/events/

tems, assuming several types of incompleteness. The second paper applies both rough-set-based [18] and fuzzy-set-based [12] rules in the task of posture detection. This is an example of real-world application, whereby the multi-stage solution needs to integrate sensor calibration, sensor data acquisition, inducing rules from the acquired data, as well as rule-based inference. With respect to making all such layers working together, this work can be compared to [43], [66], [67].

Papers [97], [98] deal with ensembles of decision models. The first paper employs so-called Dominance-based Rough Set Balanced Rule Ensemble for fraud detection. Herein, it is worth adding that rough set methods and applications include also examples of operating with ensembles of the aforementioned approximate reducts [99] and bireducts [17], [58] which correspond to bigger collections of rules. The second paper shows how to negotiate between classifiers and actually refers to the aforementioned conflict analysis model proposed by Professor Pawlak [6], [100]. On the other hand, the mechanism of voting in the third paper relies on the aforementioned three-way decision making [21]. It is worth emphasizing that solutions described in both papers attempt to provide a deeper insight into the ensemble decisions.

Paper [101] remains in the area of ensembles of decision models but it also touches an important aspect of incremental learning in dynamic data environments [102]. Herein, it is worth recalling a gentle difference between reasoning about objects or states in a repetitive fashion (whereby the values of attributes in information systems need to be cyclically updated) and reasoning about temporal objects or phenomena (which require different construction of information systems with attributes reflecting changes and trends) [103], [104].

Finally, papers [105], [106] combine the principles of rough sets and GrC with popular machine learning methods, referring to decision model ensembles as well. The idea is to prepare compacted data inputs – called granular reflections [15] – for the algorithms responsible for learning decision models such as *e.g.* neural networks or random forests (see [93] again). From a conceptual perspective, it corresponds to the aforementioned studies on aggregated / granulated / summarized information systems [63], [66]. This topic has also interesting relationships with some branches of approximate computing [107] and compressed image recognition [108].

III. INFORMATION SYSTEMS AND IGrC

As we have already emphasized, rough sets are based on data / information granulation. Both the original rough set approach and its extensions, need granules (and their descriptions / signatures) as inputs to compute approximations. The same applies to rough set methods of constructing decision models, *e.g.* rule-based models [52], [95]. On the other hand, granules can take different forms and have different origins. They can be partition blocks (induced by combinations of attribute values or ranges), dominance classes or neighborhoods [20], [89], relationships based on fuzzy (dis)similarity and (in)discernibility [78], [80] and so on. In information systems, granules can take different information signatures such as

precise values, value sets, ranges and distributions [35], [57]. Those signatures can be computed using different aggregation mechanisms, often assuming non-trivial interdependencies with processes and devices that produce the data [63], [66]. The reliability and accessibility of information – therefore also reliability and accessibility of the outcomes of calculations over information systems – requires a careful analysis of all phases of forming the contents of such systems.

In Section I, we highlighted that the IGrC framework [37], [40] could be helpful to keep information systems aligned with respect to practical needs of operations on them in different contexts. In the next subsections, we will elaborate on several aspects of such alignment. As already discussed, IGrC uses so-called *c*-granules in order to create configurations of physical objects and control interactions between them so as to achieve computational objectives. Now let us add that the control mechanisms embedded within *c*-granules rely on one more type of granules – informational *c*-granules (*ic*-granules) which include both abstract (informational) and physical layers. They contain specifications how to link the abstract and physical worlds, whereby the abstract world corresponds in particular to (the networks of) information systems. The perceived properties of physical objects can be used to transform the current configurations of *ic*-granules, *i.e.* to modify interactions between objects. Such mechanisms require a design of new methods of reasoning about *where*, *when*, *what*, and *how* to perceive using different sensors or actuators. New methods for judging membership (alignment, matching) of the perceived situations in (with) rough set approximations of complex concepts are needed too.

A. Reliability of Information

This kind of reliability is studied in many fields. In the domain of big data, it is referred as one of the “V’s” – Veracity [109]. Actually, we have already dealt with some of other “V’s” in the previous sections, *e.g.* Volume [61], Velocity [101] and Variety [60]. However, without addressing Veracity, *i.e.* assuring data quality that is transformed into information reliability, any solutions focused on those other “V’s” cannot guarantee anything useful. Another popular term related to this problem is “garbage data”. It refers to the fact that if a machine learning method is executed on improper data, then the resulting models cannot be expected to work successfully. The causes of data being garbage data may be connected to problems with *e.g.* sensor measurements, data parsing, or even data labels acquired from human experts [67], [110].

A technical solution to cope with garbage data is often to filter them out by using validation procedures (*e.g.* checking sensor scales) [99]. However, in many applications – such as [66] (Subsection II-A), [96] (Subsection II-B) or just-mentioned [67] – it would lead toward disqualifying too broad data fragments, if any formalized validation is possible at all. Another approach is to *live* with the unreliable data and moreover, to take such unreliability into account while conducting any computations. With this respect, non-deterministic information systems have some tools to express uncertainty

by replacing precise values with sets, intervals, etc. [35], [63]. However, (a degree of) reliability remains something different, as it refers to the way the data was acquired from the physical world rather than the specification of attribute values.

The IGrC framework is quite natural when it comes to reasoning about such an additional layer of information. Interactive granular computations can be actually extended toward adaptive searching strategies for the most relevant and reliable data, spatio-temporal windows pointing out to fragments of the physical world where the most reliable measurements and / or actions should be performed, and so on. This kind of reasoning may be also associated with the domain of data governance which extends towards data and information security, accessibility, as well as the protocols of interactions between intelligent systems and humans [111], [112]. On the other hand, the discussed physical-world-related aspects can be an additional contribution of IGrC to data governance. Moreover, IGrC can be helpful to operate with often softly expressed regulations about data integrity and timeliness.

It is also worth referring the above discussion to the meanings of aleatoric and epistemic uncertainties in machine learning [113]. From this perspective, a limited reliability of the contents of information systems can be treated as one of ingredients of the epistemic uncertainty, as it puts together both, the model and the data deficiencies. However, we believe that these two sources of deficiencies should be kept separately, with the third type of experimental / physical uncertainty explicitly considered. The analysis of this third type of uncertainty should be taken into account when assessing the efficiency and stability of machine learning models, especially given the fact that in some practical scenarios the inputs to the learning algorithms can be unreliably extracted for the purpose of *e.g.* accelerating computations [114].

B. Acquisition of Information

In order to talk about information reliability, we first need to assure that information can be gathered at all. In practice, there is often a great variety of data available but it does not mean yet that the corresponding information is sufficiently complete to perform any kind of analysis. (This relates to one more “V”: Value.) Some promising approaches to data enrichment refer to the paradigm of active learning [115], which can be further extended toward establishing an interactive loop within which subject matter experts label data objects that are of the highest interest to the machine learning algorithms. One just needs to think about controlling the quality of such labels [110].

Similarly, the data enrichment processes can rely on connecting information systems with physical systems [116]. Actually, it is worth pointing out that humans can be considered as a special kind of physical objects that interact with decision support systems and / or intelligent systems. This refers to a broader topic of the information and communication technologies (ICT) systems [39] which put together the aspects of hardware (*e.g.* sensors), software (*e.g.* machine learning methods), the data (including domain knowledge), and the system users (in particular subject matter experts).

The above ideas require a firm layer that connects information systems with the physical world where the data comes from. In IGrC, every granule should have an access to instructions how to compose the values of particular attributes for particular objects [38], [40]. Moreover, it is important for this layer to log a history of attempts to calculate particular fragments of an information system. Such history may let us avoid mistakes and misinterpretations related to the data acquisition processes. That history may be also useful while assessing reliability of the current contents of an information system. Such mechanisms can be adopted also from the architectures of granular database engines [62], [107], whereby the aspects of information completeness and reliability are equally important as in the field of machine learning.

Going back to the framework of active learning, let us claim that subject matter experts can assist us not only in enriching the data with labels but also enriching the data mining algorithms with domain insights. As an example, let us think one more time about the task of feature selection. There are various techniques of measuring and visualizing attribute importance [88], [90], [94] but they are usually applied to report to humans the final results instead of “inviting” them into a more interactive dialogue on feature selection process. In this regard, we refer to [85] where incrementally constructed information systems are employed to guide subject matter experts through such an interactive process, letting them share their recommendations about the most relevant attributes.

Last but not least, when it comes to decision problems related to complex phenomena, it is worth attempting – using the elements of active learning and human-computer interaction – to acquire from subject matter experts even more advanced knowledge, expressed in terms of hierarchical structures and dependencies. This fits the paradigm of computing with words [117] (which also corresponds to the foundations of information granulation with respect to decomposing complex problems onto their smaller components) and, in particular, the following challenge formulated by Judea Pearl [118]:

Traditional statistics is strong in devising ways of describing data and inferring distributional parameters from sample. Causal inference requires two additional ingredients: a science-friendly language for articulating causal knowledge, and a mathematical machinery for processing that knowledge, combining it with data and drawing new causal conclusions about a phenomenon.

C. Accessibility and Cost of Information

In practical deployments, there is always a risk that some of data sources – which are needed to calculate some of attributes being inputs to a decision model – will be temporarily unavailable (because of *e.g.* physical connection problems or dissatisfaction of some data governance rules) or unreliable (as discussed in Subsection III-A). Feature selection [56], including contribution of rough set methods to elimination / reduction of redundant attributes [10], can be a remedium to this problem – less attributes require less aspects of data to be

calculated. Moreover, it is possible to diversify data sources needed to derive attributes that are used by particular models in an ensemble [17], [55]. This increases a chance that at least some of models would be usable in a given situation.

However, it is not only about the accessibility, sometimes it is also about the cost. For instance, in the recent contests organized at one of online data mining competition platforms¹⁰ [119], [120], the participants purposefully did not take into account some of the available data sources (modalities) because – according to them – derivation of attributes from those sources would be too expensive computationally. Going further along this path, one may say that even if such “expensive” attributes are included into a model, its deployed version should have a choice to decide dynamically which of them (and at what level of precision [114]) are necessary to be calculated. This is important the more so as in intelligent systems dealing with complex phenomena, the most adequate selections and meanings of attributes can be changed over time [40].

Going even further, if appropriate metadata is maintained on the side of an information system, the same attribute values can be derived from different information sources or using different data modalities, perhaps subject to different cost and precision [43], [67]. This is well-aligned with the IGrC assumptions discussed in the previous subsections. The point is to pass to information granules the decision power in regard to how they produce information that they are responsible for, and make decision models responsible for timely asking those granules for particular information pieces [38], [116].

The above discussion can be extended toward a broader topic of whether the data / information updates should be rather “pushed” or “pulled” in the computational pipelines which involve learning and applying the learnt models. A common assumption is that any change in the underlying information should be more or less quickly transmitted to the inputs of a model, causing its recalculation or at least modification of its behavior [101], [102]. However, in big data scenarios it is not so obvious – it may be safer to leave such decisions in hands of information granules equipped with well-designed triggers and internal cost models [37], [107].

D. Networks of Information Systems

Continuing with the topic of intelligent systems aimed at reasoning about complex phenomena, we already know that data sources required to learn the underlying decision models cannot be acquired in a single-step process. It is necessary to provide such systems with permanent links to relevant fragments of the physical world and keep adapting (actively but also reasonably, from the computational perspective) the induced models following changes in the perceived situation. Recalling our comments about logical reasoning related to hierarchical systems [43] (see the end of Section I), one should also be aware that these hierarchical structures are dynamically changing in time and the relevant reasoning methods should allow to the system to perform the necessary reasoning about

such dynamical structures. This seems to be aligned with the following opinion expressed by Frederick Brooks [121]:

Mathematics and the physical sciences made great strides for three centuries by constructing simplified models of complex phenomena, deriving, properties from the models, and verifying those properties experimentally. This worked because the complexities ignored in the models were not the essential properties of the phenomena. It does not work when the complexities are the essence.

The starting point is to work with environments which create, maintain and synchronize multiple dynamic information systems. Such networks of information systems would be still a kind of abstraction of the real world but on the other hand, they would reflect it more accurately than single systems.

Let us first focus on the aforementioned hierarchical systems. We have already referred to the approaches whereby domain knowledge – expressed in terms of ontologies of concepts associated with a particular decision problem – is utilized to decompose that problem onto simpler components located within a hierarchical schema and then, to aggregate perceived information along that schema [45], [117]. To facilitate such aggregation process, it is indeed convenient to design a hierarchy of information systems whose objects (and therefore also attributes) correspond to different levels of conceptual granularity. This idea is actually analogous to modeling the data by means of multi-table relational database structures [114], [119], and it can be observed in quite a few applications mentioned earlier [46], [66], [67], [103].

Somewhat “orthogonal” aspect of thinking about multiple information systems refers to concurrency and distributed computations. From this perspective, at each level of the above-discussed hierarchies, we may actually imagine a group of systems working collectively and exchanging information. Herein, it is important to refer to the models proposed by Professor Pawlak [7], as well as the history of the aforementioned conferences on Concurrency, Specification and Programming (CS&P). Furthermore, it is useful to refer also to the works on the networks of information systems linked by so-called infomorphisms [32], [122]. Some relevant realizations can be found also in other domains. For instance, the already-considered granular database engine [107] contained a mechanism of distributed execution of analytical queries, whereby particular computational nodes could exchange with each other some approximate partial answers and, basing on such understood rough set approximations, decide autonomously whether it is worth requesting for the precise results.

Once we have a hierarchy / distribution of information systems, we can extend their network with the IGrC-based connections to the physical world [38], [123]. This implies a number of challenges, as the above-discussed coordination between particular information systems needs to be combined with coordination of each single system with its physical “alter ego”. For instance, we can consider a more active version of the tasks of attribute selection and extraction [56], [60], whereby it is required to develop new methods of

¹⁰knowledgepit.ai

selection and construction of sensors. At a more general level, the whole idea requires a distributed control of *c*-granules, whereby specific reasoning methods (related to cooperation / competition between granules) need to express the expected behavioral patterns of the whole “society” of granules. Herein, one can seek for inspirations in the previously discussed conflict analysis [6]. The requirements of the aforementioned ICT systems [39], web intelligence [65] or *e.g.* IoT analytics [124] – whereby there are a number of federated learning scenarios involving distributed agents (and their underlying information systems) – can be a useful analogy as well.

IV. CONCLUSIONS AND FUTURE DIRECTIONS

The first goal of this paper was to expose the current progress of the theory and applications of rough sets – the methodology founded by Zdzisław Pawlak with the aim of deriving and expressing important patterns and dependencies subject to limited (incomplete, imprecise) information about the concepts of practical interest [1], [9]. We examined connections of rough sets with decision making [20], [21], logics [73], [76], probability [47], [72], statistical / machine learning [77], [105], data mining [19], [23], fuzzy sets [13], [48], formal concept analysis [26], [69], and other data / information / knowledge representation methodologies [31], [32]. We discussed some of rough set techniques aimed at attribute selection / reduction treated as a component of knowledge discovery processes [10], [17], with particular emphasis on computational scalability challenges [60], [82]. We paid special attention to rough set approaches to construction of interpretable (explainable by design) rule-based decision models [18], [48], [52], [97]. We referred to rough set software packages for data mining and machine learning [24], [68], [83], as well as other technologies which utilize rough set approximation principles for their internal purposes [62]. We also recalled several (out of many) applications of rough set methods in real-world data analysis, including biomedical and healthcare applications whereby interpretability of decision models is of special importance [66], [67], [84], [91].

Our second goal was to address the progress in the area of information systems [3], [4]. We referred to their extensions [33], [70] and we outlined a number of applications which use specifically formed information systems as the means for representing (granulated / aggregated) data, (uncertain / imprecise) information, and (appropriately transformed) knowledge structures [35], [43], [63], [89]. We pointed out that information systems – especially their hierarchies and networks – constitute the means for reasoning about complex spatio-temporal phenomena [45], [104]. We also claimed that information systems can be a medium to conduct interactive data analytics involving subject matter experts [85] and support interactions between multiple data exploration processes [123]. That led us toward discussing the current challenges (often referred as the big data “V’s”) in front of information systems understood as the means for representing and delivering data required for the learning processes [102], [109], [110], [119]. Accordingly, we examined whether the principles of so-called interactive

granular computing (IGrC) [37], [116] can help us to face those challenges and to what extent they are aligned with some of emerging trends in machine learning [115], [124].

It was important for us to discuss the principles of granular computing – including IGrC – together with rough sets and information systems, as these three domains interfere with each other in many interesting ways [11], [14], [15], [40]. In particular, IGrC may have future implications for the design of intelligent systems, *e.g.* when it comes to so-called perceptual rough sets¹¹. If one wants to build rough set approximations of complex concepts in real-world environments, then it is required to design a dynamic space of granules which are able to reason about complex approximation constructions. The corresponding reasoning methods will need to be far richer than the ones considered so far in rough set applications.

Some other future directions for rough sets and information systems refer to continuation of development of real-world applications, focused on *e.g.* images and video recordings [22], [46], [120], as well as signals and sensor measurements [99], [96], [103]. This kind of development should emphasize strong assets of rough sets, such as straightforward interpretability of the derived decision models, even when it comes to modeling very complex and dynamic situations [11], [101]. Needless to say, interpretability is now the key objective for a great majority of machine learning applications [87], [88].

When thinking about the future it is also worth referring to the history. That reflected one more objective: acknowledging the 40th anniversary of rough sets [1], their founder [5], as well as some of relevant past and present events such as FedCSIS 2012 (rough set papers published exactly 10 years ago) [47], [48], [51], [53], [54], [57], [58], [59], [61], [64], [66], the PP-RAI 2022 rough set contest [69], [70], [72], [73], [77], [78], [81], [82], [83], [84], [89], [90], [91], [95], [96], [97], [98], [101], [105], [106] and celebration of the 30th CS&P – the event series whereby this paper’s topics have been regularly addressed [36], [40], [44], [103], [116], [123].

In the end, let us recall that this is not the first anniversary corresponding to rough sets in the history of the FedCSIS conferences. Indeed, FedCSIS 2016 (Gdańsk, Poland) hosted the international panel discussion in memoriam of the 90th anniversary of the birth and the 10th anniversary of the death of Professor Pawlak¹². The previously-cited publications [4], [8], [100] were prepared specially for that panel.

REFERENCES

- [1] Z. Pawlak, “Rough Sets,” *International Journal of Computer and Information Sciences*, vol. 11, no. 5, pp. 341–356, 1982. [Online]. Available: doi.org/10.1007/BF01001956
- [2] —, *Rough Sets – Theoretical Aspects of Reasoning about Data*, ser. Theory and Decision Library D. Springer, 1991. [Online]. Available: doi.org/10.1007/978-94-011-3534-4

¹¹See *e.g.* A. Skowron: Perceptual Rough Set Approach in Interactive Granular Computing (keynote). Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 2022), July 11-15, 2022, Milan, Italy. roughsets.org/bin/1f094938771802d02dec99c66823875c.PDF

¹²fedcsis.org/2016/plenary_panel

- [3] —, “Information Systems – Theoretical Foundations,” *Information Systems*, vol. 6, no. 3, pp. 205–218, 1981. [Online]. Available: doi.org/10.1016/0306-4379(81)90023-5
- [4] V. W. Marek, “Working with Zdzisław Pawlak – Personal Reminiscences,” in *Proceedings of the 2016 Federated Conference on Computer Science and Information Systems, FedCSIS 2016, Gdańsk, Poland, September 11-14, 2016*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 8. IEEE, 2016, pp. 189–190. [Online]. Available: doi.org/10.15439/2016F002
- [5] A. Skowron, M. K. Chakraborty, J. W. Grzymała-Busse, V. W. Marek, S. K. Pal, J. F. Peters, G. Rozenberg, D. Ślęzak, R. Słowiński, S. Tsumoto, A. Wakulicz-Deja, G. Wang, and W. Ziarko, “Professor Zdzisław Pawlak (1926-2006): Founder of the Polish School of Artificial Intelligence,” in *Rough Sets and Intelligent Systems – Professor Zdzisław Pawlak in Memoriam – Volume 1*, ser. Intelligent Systems Reference Library, A. Skowron and Z. Suraj, Eds. Springer, 2013, vol. 42, pp. 1–56. [Online]. Available: doi.org/10.1007/978-3-642-30344-9_1
- [6] Z. Pawlak, “An Inquiry into Anatomy of Conflicts,” *Journal of Information Sciences*, vol. 109, pp. 65–78, 1998. [Online]. Available: doi.org/10.1016/S0020-0255(97)10072-X
- [7] —, “Concurrent versus Sequential – the Rough Sets Perspective,” *Bulletin of the EATCS*, vol. 48, pp. 178–190, 1992.
- [8] J. F. Peters and S. Ramanna, “Maximal Nucleus Clusters in Pawlak Paintings. Nerves as Approximating Tools in Visual Arts,” in *Proceedings of the 2016 Federated Conference on Computer Science and Information Systems, FedCSIS 2016, Gdańsk, Poland, September 11-14, 2016*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 8. IEEE, 2016, pp. 199–202. [Online]. Available: doi.org/10.15439/2016F004
- [9] Z. Pawlak and A. Skowron, “Rudiments of Rough Sets,” *Information Sciences*, vol. 177, no. 1, pp. 3–27, 2007. [Online]. Available: doi.org/10.1016/j.ins.2006.06.003
- [10] —, “Rough Sets and Boolean Reasoning,” *Information Sciences*, vol. 177, no. 1, pp. 41–73, 2007. [Online]. Available: doi.org/10.1016/j.ins.2006.06.007
- [11] —, “Rough Sets: Some Extensions,” *Information Sciences*, vol. 177, no. 1, pp. 28–40, 2007. [Online]. Available: doi.org/10.1016/j.ins.2006.06.006
- [12] L. A. Zadeh, “Fuzzy Sets,” *Information and Control*, vol. 8, no. 3, pp. 338–353, 1965. [Online]. Available: doi.org/10.1016/S0019-9958(65)90241-X
- [13] C. Cornelis, “Hybridization of Fuzzy Sets and Rough Sets: Achievements and Opportunities,” in *Proceedings of the 2022 Federated Conference on Computer Science and Intelligence Systems, Sofia, Bulgaria, September 4-7, 2022*, ser. Annals of Computer Science and Information Systems, M. Ganzha, M. Paprzycki, and D. Ślęzak, Eds., vol. 30, 2022.
- [14] S. K. Pal, “Soft Data Mining, Computational Theory of Perceptions, and Rough-Fuzzy Approach,” *Information Sciences*, vol. 163, no. 1-3, pp. 5–12, 2004. [Online]. Available: doi.org/10.1016/j.ins.2003.03.014
- [15] L. Polkowski and P. Artiemjew, *Granular Computing in Decision Approximation – An Application of Rough Mereology*, ser. Intelligent Systems Reference Library. Springer, 2015, vol. 77. [Online]. Available: doi.org/10.1007/978-3-319-12880-1
- [16] A. Skowron and J. Stepaniuk, “Information Granules: Towards Foundations of Granular Computing,” *International Journal of Intelligent Systems*, vol. 16, no. 1, pp. 57–85, 2001.
- [17] S. Stawicki, D. Ślęzak, A. Janusz, and S. Widz, “Decision Bireducts and Decision Reducts – A Comparison,” *International Journal of Approximate Reasoning*, vol. 84, pp. 75–109, 2017. [Online]. Available: doi.org/10.1016/j.ijar.2017.02.007
- [18] J. W. Grzymała-Busse, “Rule Induction,” in *Data Mining and Knowledge Discovery Handbook, 2nd ed.*, O. Maimon and L. Rokach, Eds. Springer, 2010, pp. 249–265. [Online]. Available: doi.org/10.1007/978-0-387-09823-4_13
- [19] H. S. Nguyen, “Approximate Boolean Reasoning: Foundations and Applications in Data Mining,” *Transactions on Rough Sets*, vol. 5, pp. 334–506, 2006. [Online]. Available: doi.org/10.1007/11847465_16
- [20] S. Greco, B. Matarazzo, and R. Słowiński, “Rough Sets Theory for Multicriteria Decision Analysis,” *European Journal of Operational Research*, vol. 129, no. 1, pp. 1–47, 2001. [Online]. Available: doi.org/10.1016/S0377-2217(00)00167-3
- [21] Y. Yao, “Three-Way Decisions and Cognitive Computing,” *Cognitive Computation*, vol. 8, no. 4, pp. 543–554, 2016. [Online]. Available: doi.org/10.1007/s12559-016-9397-5
- [22] P. Maji and S. K. Pal, “Maximum Class Separability for Rough-Fuzzy C-Means Based Brain MR Image Segmentation,” *Transactions on Rough Sets*, vol. 9, pp. 114–134, 2008. [Online]. Available: doi.org/10.1007/978-3-540-89876-4_7
- [23] G. Peters, F. A. Crespo, P. Lingras, and R. Weber, “Soft Clustering – Fuzzy and Rough Approaches and Their Extensions and Derivatives,” *International Journal of Approximate Reasoning*, vol. 54, no. 2, pp. 307–322, 2013. [Online]. Available: doi.org/10.1016/j.ijar.2012.10.003
- [24] L. S. Riza, A. Janusz, C. Bergmeir, C. Cornelis, F. Herrera, D. Ślęzak, and J. M. Benítez, “Implementing Algorithms of Rough Set Theory and Fuzzy Rough Set Theory in the R Package ‘RoughSets,’” *Information Sciences*, vol. 287, pp. 68–89, 2014. [Online]. Available: doi.org/10.1016/j.ins.2014.07.029
- [25] R. Wille, “Restructuring Lattice Theory: An Approach Based on Hierarchies of Concepts,” in *Ordered Sets, Proceedings*, ser. NATO Advanced Studies Institute, I. Rival, Ed., vol. 83. Dordrecht: Springer, 1982, pp. 445–470.
- [26] R. E. Kent, “Rough Concept Analysis: A Synthesis of Rough Sets and Formal Concept Analysis,” *Fundam. Informaticae*, vol. 27, no. 2/3, pp. 169–181, 1996. [Online]. Available: doi.org/10.3233/FI-1996-272305
- [27] S. Naouali and R. Missaoui, “Flexible Query Answering in Data Cubes,” in *Data Warehousing and Knowledge Discovery, 7th International Conference, DaWaK 2005, Copenhagen, Denmark, August 22-26, 2005, Proceedings*, ser. Lecture Notes in Computer Science, A. M. Tjoa and J. Trujillo, Eds., vol. 3589. Springer, 2005, pp. 221–232. [Online]. Available: doi.org/10.1007/11546849_22
- [28] D. Ślęzak and V. Eastwood, “Data Warehouse Technology by Infobright,” in *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2009, Providence, Rhode Island, USA, June 29 – July 2, 2009*, U. Çetintemel, S. B. Zdonik, D. Kossmann, and N. Tatbul, Eds. ACM, 2009, pp. 841–846. [Online]. Available: doi.org/10.1145/1559845.1559933
- [29] M. L. Hughes, R. M. Shank, and E. S. Stein, *Decision Tables*. MDI Publications, 1968.
- [30] A. M. Moreno Garcia, M. Verhelle, and J. Vanthienen, “An Overview of Decision Table Literature 1982-2000,” in *The Fifth International Conference on Artificial Intelligence and Emerging Technologies in Accounting, Finance and Tax, Huelva, Spain, November 2-3, 2000*. [Online]. Available: feb.kuleuven.be/prologal/download/overview82-2000.pdf
- [31] I. Chikalov, V. V. Lozin, I. Lozina, M. Moshkov, H. S. Nguyen, A. Skowron, and B. Zielosko, *Three Approaches to Data Analysis – Test Theory, Rough Sets and Logical Analysis of Data*, ser. Intelligent Systems Reference Library. Springer, 2013, vol. 41. [Online]. Available: doi.org/10.1007/978-3-642-28667-4
- [32] A. Skowron, J. Stepaniuk, and J. F. Peters, “Rough Sets and Infomorphisms: Towards Approximation of Relations in Distributed Environments,” *Fundamenta Informaticae*, vol. 54, no. 2-3, pp. 263–277, 2003. [Online]. Available: content.iospress.com/articles/fundamenta-informaticae/fi54-2-3-12
- [33] E. Orłowska and Z. Pawlak, “Representation of Nondeterministic Information,” *Theoretical Computer Science*, vol. 29, pp. 27–39, 1984. [Online]. Available: doi.org/10.1016/0304-3975(84)90010-0
- [34] W. Lipski Jr., “On Databases with Incomplete Information,” *Journal of the ACM*, vol. 28, no. 1, pp. 41–70, 1981. [Online]. Available: doi.org/10.1145/322234.322239
- [35] H. Sakai and M. Nakata, “Rough Set-based Rule Generation and Apriori-based Rule Generation from Table Data Sets: A Survey and a Combination,” *CAAI Transactions on Intelligence Technology*, vol. 4, no. 4, pp. 203–213, 2019. [Online]. Available: doi.org/10.1049/trit.2019.0001
- [36] M. Wolski and A. Gomolińska, “Semantic Rendering of Data Tables – Multivalued Information Systems Revisited,” in *Proceedings of the 25th International Workshop on Concurrency, Specification and Programming, Rostock, Germany, September 28-30, 2016*, ser. CEUR Workshop Proceedings, B. Schlingloff, Ed., vol. 1698. CEUR-WS.org, 2016, pp. 113–124. [Online]. Available: ceur-ws.org/Vol-1698/CS&P2016_11_Wolski&Gomolinska_Semantic-Rendering-of-Data-Tables-Multivalued-Information-Systems-Revisited.pdf
- [37] A. Skowron and A. Jankowski, “Rough Sets and Interactive Granular

- Computing,” *Fundamenta Informaticae*, vol. 147, no. 2-3, pp. 371–385, 2016. [Online]. Available: doi.org/10.3233/FI-2016-1413
- [38] A. Skowron and P. Wasilewski, “Interactive Information Systems: Toward Perception Based Computing,” *Theoretical Computer Science*, vol. 454, pp. 240–260, 2012. [Online]. Available: doi.org/10.1016/j.tcs.2012.04.019
- [39] M. Lippi, S. Mariani, M. Martinelli, and F. Zambonelli, “Individual and Collective Self-Development: Concepts and Challenges,” in *Proceedings of the 2022 Federated Conference on Computer Science and Intelligence Systems, Sofia, Bulgaria, September 4-7, 2022*, ser. Annals of Computer Science and Information Systems, M. Ganzha, M. Paprzycki, and D. Ślęzak, Eds., vol. 30, 2022.
- [40] A. Jankowski, A. Skowron, and R. W. Świniarski, “Interactive Complex Granules,” in *Proceedings of the 22nd International Workshop on Concurrency, Specification and Programming, Warsaw, Poland*, ser. CEUR Workshop Proceedings, M. S. Szczuka, L. Czaja, and M. Kacprzak, Eds., vol. 1032. CEUR-WS.org, 2013, pp. 206–218. [Online]. Available: ceur-ws.org/Vol-1032/paper-18.pdf
- [41] T. A. Poggio and S. Smale, “The Mathematics of Learning: Dealing with Data,” *Notices of the American Mathematical Society*, vol. 50, no. 5, pp. 537–544, 2003.
- [42] P. Stone, *Layered Learning in Multi-Agent Systems: A Winning Approach to Robotic Soccer*. MIT Press, 2000.
- [43] P. Doherty, W. Łukaszewicz, A. Skowron, and A. Szalas, *Knowledge Representation Techniques – A Rough Set Approach*, ser. Studies in Fuzziness and Soft Computing. Springer, 2006, vol. 202. [Online]. Available: doi.org/10.1007/3-540-33519-6
- [44] S. Dutta and P. Wasilewski, “Dialogue in Hierarchical Learning of a Concept Using Prototypes and Counterexamples,” in *Proceedings of the 24th International Workshop on Concurrency, Specification and Programming, Rzeszów, Poland, September 28-30, 2015*, ser. CEUR Workshop Proceedings, Z. Suraj and L. Czaja, Eds., vol. 1492. CEUR-WS.org, 2015, pp. 126–133. [Online]. Available: ceur-ws.org/Vol-1492/Paper_12.pdf
- [45] J. G. Bazan, “Hierarchical Classifiers for Complex Spatio-Temporal Concepts,” *Transactions on Rough Sets*, vol. 9, pp. 474–750, 2008. [Online]. Available: doi.org/10.1007/978-3-540-89876-4_26
- [46] S. H. Nguyen, T. T. Nguyen, M. S. Szczuka, and H. S. Nguyen, “An Approach to Pattern Recognition Based on Hierarchical Granular Computing,” *Fundamenta Informaticae*, vol. 127, no. 1-4, pp. 369–384, 2013. [Online]. Available: doi.org/10.3233/FI-2013-915
- [47] I. Düntsch and G. Gediga, “Weighted Lambda Precision Models in Rough Set Data Analysis,” in *Federated Conference on Computer Science and Information Systems – FedCSIS 2012, Wrocław, Poland, 9-12 September 2012, Proceedings*, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., 2012, pp. 287–294. [Online]. Available: fedcsis.org/proceedings/2012/pliks/89.pdf
- [48] T. Fan, C. Liau, and D. Liu, “Variable Precision Fuzzy Rough Set Based on Relative Cardinality,” in *Federated Conference on Computer Science and Information Systems – FedCSIS 2012, Wrocław, Poland, 9-12 September 2012, Proceedings*, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., 2012, pp. 43–47. [Online]. Available: fedcsis.org/proceedings/2012/pliks/398.pdf
- [49] W. Ziarko, “Variable Precision Rough Set Model,” *Journal of Computer and System Sciences*, vol. 46, no. 1, pp. 39–59, 1993. [Online]. Available: doi.org/10.1016/0022-0000(93)90048-2
- [50] Z. Pawlak, “Rough Sets, Decision Algorithms and Bayes’ Theorem,” *European Journal of Operational Research*, vol. 136, no. 1, pp. 181–189, 2002. [Online]. Available: doi.org/10.1016/S0377-2217(01)00029-7
- [51] B. Zielosko, “Sequential Optimization of γ -Decision Rules,” in *Federated Conference on Computer Science and Information Systems – FedCSIS 2012, Wrocław, Poland, 9-12 September 2012, Proceedings*, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., 2012, pp. 339–346. [Online]. Available: fedcsis.org/proceedings/2012/pliks/87.pdf
- [52] M. Kryszkiewicz, “ACBC-Adequate Association and Decision Rules Versus Key Generators and Rough Sets Approximations,” *Fundamenta Informaticae*, vol. 148, no. 1-2, pp. 65–85, 2016. [Online]. Available: doi.org/10.3233/FI-2016-1423
- [53] L. G. Nguyen, “Metric Based Attribute Reduction in Decision Tables,” in *Federated Conference on Computer Science and Information Systems – FedCSIS 2012, Wrocław, Poland, 9-12 September 2012, Proceedings*, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., 2012, pp. 311–316. [Online]. Available: fedcsis.org/proceedings/2012/pliks/311.pdf
- [54] L. G. Nguyen and H. S. Nguyen, “On Elimination of Redundant Attributes from Decision Table,” in *Federated Conference on Computer Science and Information Systems – FedCSIS 2012, Wrocław, Poland, 9-12 September 2012, Proceedings*, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., 2012, pp. 317–322. [Online]. Available: fedcsis.org/proceedings/2012/pliks/324.pdf
- [55] R. Polikar, J. DePasquale, H. S. Mohammed, G. Brown, and L. I. Kuncheva, “Learn⁺⁺.MF: A Random Subspace Approach for the Missing Feature Problem,” *Pattern Recognition*, vol. 43, no. 11, pp. 3817–3832, 2010. [Online]. Available: doi.org/10.1016/j.patcog.2010.05.028
- [56] M. Dash and H. Liu, “Consistency-based Search in Feature Selection,” *Artificial Intelligence*, vol. 151, no. 1-2, pp. 155–176, 2003. [Online]. Available: doi.org/10.1016/S0004-3702(03)00079-1
- [57] M. Azad, I. Chikalov, M. Moshkov, and B. Zielosko, “Tests for Decision Tables with Many-Valued Decisions – Comparative Study,” in *Federated Conference on Computer Science and Information Systems – FedCSIS 2012, Wrocław, Poland, 9-12 September 2012, Proceedings*, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., 2012, pp. 271–277. [Online]. Available: fedcsis.org/proceedings/2012/pliks/140.pdf
- [58] S. Stawicki and S. Widz, “Decision Bireducts and Approximate Decision Reducts: Comparison of Two Approaches to Attribute Subset Ensemble Construction,” in *Federated Conference on Computer Science and Information Systems – FedCSIS 2012, Wrocław, Poland, 9-12 September 2012, Proceedings*, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., 2012, pp. 331–338. [Online]. Available: fedcsis.org/proceedings/2012/pliks/348.pdf
- [59] A. Janusz and D. Ślęzak, “Utilization of Attribute Clustering Methods for Scalable Computation of Reducts from High-Dimensional Data,” in *Federated Conference on Computer Science and Information Systems – FedCSIS 2012, Wrocław, Poland, 9-12 September 2012, Proceedings*, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., 2012, pp. 295–302. [Online]. Available: fedcsis.org/proceedings/2012/pliks/330.pdf
- [60] M. Grzegorowski, A. Janusz, D. Ślęzak, and M. S. Szczuka, “On the Role of Feature Space Granulation in Feature Selection Processes,” in *2017 IEEE International Conference on Big Data (IEEE BigData 2017), Boston, MA, USA, December 11-14, 2017*, J. Nie, Z. Obradovic, T. Suzumura, R. Ghosh, R. Nambiar, C. Wang, H. Zang, R. Baeza-Yates, X. Hu, J. Kepner, A. Cuzzocrea, J. Tang, and M. Toyoda, Eds. IEEE Computer Society, 2017, pp. 1806–1815. [Online]. Available: doi.org/10.1109/BigData.2017.8258124
- [61] M. Kowalski and S. Stawicki, “SQL-based Heuristics for Selected KDD Tasks over Large Data Sets,” in *Federated Conference on Computer Science and Information Systems – FedCSIS 2012, Wrocław, Poland, 9-12 September 2012, Proceedings*, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., 2012, pp. 303–310. [Online]. Available: fedcsis.org/proceedings/2012/pliks/395.pdf
- [62] M. Wnuk, S. Stawicki, and D. Ślęzak, “Reinventing Infobright’s Concept of Rough Calculations on Granulated Tables for the Purpose of Accelerating Modern Data Processing Frameworks,” in *2020 IEEE International Conference on Big Data (IEEE BigData 2020), Atlanta, GA, USA, December 10-13, 2020*, X. Wu, C. Jermaine, L. Xiong, X. Hu, O. Kotevska, S. Lu, W. Xu, S. Aluru, C. Zhai, E. Al-Masri, Z. Chen, and J. Saltz, Eds. IEEE, 2020, pp. 5405–5412. [Online]. Available: doi.org/10.1109/BigData50022.2020.9378233
- [63] D. Ślęzak, P. Synak, A. Wojna, and J. Wróblewski, “Two Database Related Interpretations of Rough Approximations: Data Organization and Query Execution,” *Fundam. Informaticae*, vol. 127, no. 1-4, pp. 445–459, 2013. [Online]. Available: doi.org/10.3233/FI-2013-920
- [64] K. Pancerz, “Dominance-based rough set approach for decision systems over ontological graphs,” in *Federated Conference on Computer Science and Information Systems – FedCSIS 2012, Wrocław, Poland, 9-12 September 2012, Proceedings*, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., 2012, pp. 323–330. [Online]. Available: fedcsis.org/proceedings/2012/pliks/366.pdf
- [65] N. Zhong, J. Ma, R. Huang, J. Liu, Y. Yao, Y. Zhang, and J. Chen, “Research Challenges and Perspectives on Wisdom Web of Things (W2T),” *The Journal of Supercomputing*, vol. 64, no. 3, pp. 862–882, 2013. [Online]. Available: doi.org/10.1007/s11227-010-0518-8
- [66] J. G. Bazan, S. Bazan-Socha, S. Buregwa-Czuma, P. W. Pardel, and B. Sokolowska, “Predicting the Presence of Serious Coronary Artery Disease Based on 24 hour Holter ECG Monitoring,” in *Federated Conference on Computer Science and Information Systems – FedCSIS 2012, Wrocław, Poland, 9-12 September 2012, Proceedings*,

- M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., 2012, pp. 279–286. [Online]. Available: fedcis.org/proceedings/2012/pliks/227.pdf
- [67] Ł. Sosnowski and J. Wróblewski, “Toward Automatic Assessment of a Risk of Women’s Health Disorders Based on Ontology Decision Models and Menstrual Cycle Analysis,” in *2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, December 15–18, 2021*, Y. Chen, H. Ludwig, Y. Tu, U. M. Fayyad, X. Zhu, X. Hu, S. Byna, X. Liu, J. Zhang, S. Pan, V. Papalexakis, J. Wang, A. Cuzzocrea, and C. Ordóñez, Eds. IEEE, 2021, pp. 5544–5552. [Online]. Available: doi.org/10.1109/BigData52589.2021.9671481
- [68] A. Wojna and R. Latkowski, “Rseslib 3: Library of Rough Set and Machine Learning Methods with Extensible Architecture,” *Transactions on Rough Sets*, vol. 21, pp. 301–323, 2019. [Online]. Available: doi.org/10.1007/978-3-662-58768-3_7
- [69] M. J. Benítez-Caballero, J. Medina, E. Ramírez-Poussa, and D. Ślęzak, “Rough-set-driven Approach for Attribute Reduction in Fuzzy Formal Concept Analysis,” *Fuzzy Sets and Systems*, vol. 391, pp. 117–138, 2020. [Online]. Available: doi.org/10.1016/j.fss.2019.11.009
- [70] M. Wolski and A. Gomolińska, “Data Meaning and Knowledge Discovery: Semantical Aspects of Information Systems,” *International Journal of Approximate Reasoning*, vol. 119, pp. 40–57, 2020. [Online]. Available: doi.org/10.1016/j.ijar.2020.01.002
- [71] R. Belohlávek and V. Vychodil, “What is a Fuzzy Concept Lattice?” in *Proceedings of the CLA 2005 International Workshop on Concept Lattices and their Applications Olomouc, Czech Republic, September 7–9, 2005*, ser. CEUR Workshop Proceedings, R. Belohlávek and V. Snásel, Eds., vol. 162. CEUR-WS.org, 2005. [Online]. Available: ceur-ws.org/Vol-162/paper4.pdf
- [72] P. Doherty and A. Szalas, “A Landscape and Implementation Framework for Probabilistic Rough Sets Using ProbLog,” *Information Sciences*, vol. 593, pp. 546–576, 2022. [Online]. Available: doi.org/10.1016/j.ins.2021.12.062
- [73] A. Grabowski, “Automated Comparative Study of Some Generalized Rough Approximations,” *Fundamenta Informaticae*, vol. 179, no. 2, pp. 165–182, 2021. [Online]. Available: doi.org/10.3233/FI-2021-2019
- [74] L. De Raedt, A. Kimmig, and H. Toivonen, “ProbLog: A Probabilistic Prolog and Its Application in Link Discovery,” in *IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6–12, 2007*, M. M. Veloso, Ed., 2007, pp. 2462–2467. [Online]. Available: ijcai.org/Proceedings/07/Papers/396.pdf
- [75] A. Grabowski, A. Kornilowicz, and A. Naumowicz, “Four Decades of Mizar – Foreword,” *Journal of Automated Reasoning*, vol. 55, no. 3, pp. 191–198, 2015. [Online]. Available: doi.org/10.1007/s10817-015-9345-1
- [76] P. Pagliani and M. K. Chakraborty, *A Geometry of Approximation – Rough Set Theory: Logic, Algebra and Topology of Conceptual Patterns*, ser. Trends in Logic. Springer, 2008. [Online]. Available: doi.org/10.1007/978-1-4020-8622-9
- [77] Y. Kusunoki, J. Błaszczyński, M. Inuiguchi, and R. Słowiński, “Empirical Risk Minimization for Dominance-based Rough Set Approaches,” *Information Sciences*, vol. 567, pp. 395–417, 2021. [Online]. Available: doi.org/10.1016/j.ins.2021.02.043
- [78] M. Palangetic, C. Cornelis, S. Greco, and R. Słowiński, “Granular Representation of OWA-based Fuzzy Rough Sets,” *Fuzzy Sets and Systems*, vol. 440, pp. 112–130, 2022. [Online]. Available: doi.org/10.1016/j.fss.2021.04.018
- [79] V. Vapnik, “Principles of Risk Minimization for Learning Theory,” in *Advances in Neural Information Processing Systems 4, [NIPS Conference, Denver, Colorado, USA, December 2–5, 1991]*, J. E. Moody, S. J. Hanson, and R. Lippmann, Eds. Morgan Kaufmann, 1991, pp. 831–838. [Online]. Available: papers.nips.cc/paper/506-principles-of-risk-minimization-for-learning-theory
- [80] L. A. Zadeh, “Toward a Theory of Fuzzy Information Granulation and its Centrality in Human Reasoning and Fuzzy Logic,” *Fuzzy Sets and Systems*, vol. 90, no. 2, pp. 111–127, 1997. [Online]. Available: [doi.org/10.1016/S0165-0114\(97\)00077-8](https://doi.org/10.1016/S0165-0114(97)00077-8)
- [81] U. Stańczyk and B. Zielosko, “Heuristic-based Feature Selection for Rough Set Approach,” *International Journal of Approximate Reasoning*, vol. 125, pp. 187–202, 2020. [Online]. Available: doi.org/10.1016/j.ijar.2020.07.005
- [82] M. Kopczyński and T. Grześ, “Hardware Rough Set Processor Parallel Architecture in FPGA for Finding Core in Big Datasets,” *Journal of Artificial Intelligence and Soft Computing Research*, vol. 11, no. 2, pp. 99–110, 2021. [Online]. Available: doi.org/10.2478/jaiscr-2021-0007
- [83] M. Garbulowski, K. Diamanti, K. Smolińska, N. Baltzer, P. Stoll, S. Bornelöv, A. Øhrn, L. Feuk, and J. Komorowski, “R.ROSETTA: An Interpretable Machine Learning Framework,” *BMC Bioinformatics*, vol. 22, no. 1, p. 110, 2021. [Online]. Available: doi.org/10.1186/s12859-021-04049-z
- [84] S. Stratmann, S. A. Yones, M. Garbulowski, J. Sun, A. Skaftason, M. Mayrhofer, N. Norgren, M. K. Herlin, C. Sundström, A. Eriksson, M. Höglund, J. Palle, J. Abrahamsson, K. Jahnukainen, M. C. Munthe-Kaas, B. Zeller, K. Pokrovskaja Tamm, L. Cavalier, J. Komorowski, and L. Holmfeldt, “Transcriptomic Analysis Reveals Proinflammatory Signatures Associated with Acute Myeloid Leukemia Progression,” *Blood Advances*, vol. 6, no. 1, pp. 152–164, 2022. [Online]. Available: doi.org/10.1182/bloodadvances.2021004962
- [85] D. Ślęzak, M. Grzegorowski, A. Janusz, and S. Stawicki, “Toward Interactive Attribute Selection with Infolattices,” in *Rough Sets – International Joint Conference, IJCRS 2017, Olsztyn, Poland, July 3–7, 2017, Proceedings, Part II*, ser. Lecture Notes in Computer Science, L. Polkowski, Y. Yao, P. Artiemjew, D. Ciucci, D. Liu, D. Ślęzak, and B. Zielosko, Eds., vol. 10314. Springer, 2017, pp. 526–539. [Online]. Available: doi.org/10.1007/978-3-319-60840-2_38
- [86] F. Sperrle, M. El-Assady, G. Guo, R. Borgo, D. H. Chau, A. Endert, and D. A. Keim, “A Survey of Human-centered Evaluations in Human-centered Machine Learning,” *Computer Graphics Forum*, vol. 40, no. 3, pp. 543–567, 2021. [Online]. Available: doi.org/10.1111/cgf.14329
- [87] P. P. Angelov and X. Gu, “Toward Anthropomorphic Machine Learning,” *Computer*, vol. 51, no. 9, pp. 18–27, 2018. [Online]. Available: doi.org/10.1109/MC.2018.3620973
- [88] S. M. Lundberg and S. Lee, “A Unified Approach to Interpreting Model Predictions,” in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4–9, 2017, Long Beach, CA, USA*, I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, Eds., 2017, pp. 4765–4774. [Online]. Available: proceedings.neurips.cc/paper/2017/hash/8a20a8621978632d76c43dfd28b67767-Abstract.html
- [89] K. Pancerz, “Rough Set Based Description of Plasmodium Propagation,” *International Journal of Unconventional Computing*, vol. 15, no. 4, pp. 287–299, 2020. [Online]. Available: oldcitypublishing.com/journals/ijuc-home/ijuc-issue-contents/ijuc-volume-15-number-4-2020/ijuc-15-4-p-287-299/
- [90] Ł. Pałkowski, M. Karolak, J. Błaszczyński, J. Krysiński, and R. Słowiński, “Structure-Activity Relationships of the Imidazolium Compounds as Antibacterials of *Staphylococcus aureus* and *Pseudomonas Aeruginosa*,” *International Journal of Molecular Sciences*, vol. 22, no. 15, 2021. [Online]. Available: mdpi.com/1422-0067/22/15/7997
- [91] M. Karolak, Ł. Pałkowski, B. Kubiak, J. Błaszczyński, R. Łunio, W. Sawicki, R. Słowiński, and J. Krysiński, “Application of Dominance-based Rough Set Approach for Optimization of Pellets Tableting Process,” *Pharmaceutics*, vol. 12, no. 11, p. 1024, 2020. [Online]. Available: doi.org/10.3390/pharmaceutics12111024
- [92] T. Y. Lin, “Neighborhood Systems: Mathematical Models of Information Granulations,” in *Proceedings of the IEEE International Conference on Systems, Man & Cybernetics: Washington, D.C., USA, 5–8 October 2003*. IEEE, 2003, pp. 3188–3193. [Online]. Available: doi.org/10.1109/ICSMC.2003.1244381
- [93] E. Frank, M. A. Hall, and I. H. Witten, “The WEKA Workbench,” Online Appendix for “Data Mining: Practical Machine Learning Tools and Techniques”, Morgan Kaufmann, Fourth Edition, 2016.
- [94] W. R. Rudnicki, M. Kierczak, J. Koronacki, and H. J. Komorowski, “A Statistical Method for Determining Importance of Variables in an Information System,” in *Rough Sets and Current Trends in Computing, 5th International Conference, RSCTC 2006, Kobe, Japan, November 6–8, 2006, Proceedings*, ser. Lecture Notes in Computer Science, S. Greco, Y. Hata, S. Hirano, M. Inuiguchi, S. Miyamoto, H. S. Nguyen, and R. Slowinski, Eds., vol. 4259. Springer, 2006, pp. 557–566. [Online]. Available: doi.org/10.1007/11908029_58
- [95] P. G. Clark, C. Gao, J. W. Grzymala-Busse, T. Mroczek, and R. Niemiec, “Complexity of Rule Sets in Mining Incomplete Data Using Characteristic Sets and Generalized Maximal Consistent Blocks,” *Logic Journal of the IGPL*, vol. 29, no. 2, pp. 124–137, 2021. [Online]. Available: doi.org/10.1093/jigpal/jzaa041

- [96] B. Pękała, T. Mroczek, D. Gil, and M. Kępski, "Application of Fuzzy and Rough Logic to Posture Recognition in Fall Detection System," *Sensors*, vol. 22, no. 4, p. 1602, 2022. [Online]. Available: doi.org/10.3390/s22041602
- [97] J. Błaszczyński, A. T. de Almeida Filho, A. Matuszyk, M. Szelaż, and R. Słowiński, "Auto Loan Fraud Detection Using Dominance-based Rough Set Approach versus Machine Learning Methods," *Expert Systems with Applications*, vol. 163, p. 113740, 2021. [Online]. Available: doi.org/10.1016/j.eswa.2020.113740
- [98] M. Przybyła-Kasperek, "Coalitions' Weights in a Dispersed System with Pawlak Conflict Model," *Group Decision and Negotiation*, vol. 3, pp. 549–591, 2020. [Online]. Available: hdl.handle.net/20.500.12128/13891
- [99] D. Ślęzak, M. Grzegorowski, A. Janusz, M. Kozielski, S. H. Nguyen, M. Sikora, S. Stawicki, and Ł. Wróbel, "A Framework for Learning and Embedding Multi-Sensor Forecasting Models into a Decision Support System: A Case Study of Methane Concentration in Coal Mines," *Information Sciences*, vol. 451–452, pp. 112–133, 2018. [Online]. Available: doi.org/10.1016/j.ins.2018.04.026
- [100] A. Wakulicz-Deja and M. Przybyła-Kasperek, "Pawlak's Conflict Model: Directions of Development," in *Proceedings of the 2016 Federated Conference on Computer Science and Information Systems, FedCSIS 2016, Gdańsk, Poland, September 11-14, 2016*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 8. IEEE, 2016, pp. 191–197. [Online]. Available: doi.org/10.15439/2016F003
- [101] Y. Zhang, D. Miao, W. Pedrycz, T. Zhao, J. Xu, and Y. Yu, "Granular Structure-based Incremental Updating for Multi-Label Classification," *Knowledge Based Systems*, vol. 189, 2020. [Online]. Available: doi.org/10.1016/j.knsys.2019.105066
- [102] D. Brzeziński, J. Stefanowski, R. Susmaga, and I. Szczęch, "On the Dynamics of Classification Measures for Imbalanced and Streaming Data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 8, pp. 2868–2878, 2020. [Online]. Available: doi.org/10.1109/TNNLS.2019.2899061
- [103] K. Pancerz, W. Paja, M. Wrzesień, and J. Warchoł, "Classification of Voice Signals through Mining Unique Episodes in Temporal Information Systems: A Rough Set Approach," in *Proceedings of the 21th International Workshop on Concurrency, Specification and Programming, Berlin, Germany, September 26-28, 2012*, ser. CEUR Workshop Proceedings, L. Popova-Zeugmann, Ed., vol. 928. CEUR-WS.org, 2012, pp. 280–291. [Online]. Available: ceur-ws.org/Vol-928/0280.pdf
- [104] A. Skowron and P. Synak, "Reasoning in Information Maps," *Fundamenta Informaticae*, vol. 59, no. 2-3, pp. 241–259, 2004. [Online]. Available: content.iospress.com/articles/fundamenta-informaticae/fi59-2-3-10
- [105] K. Ropiak and P. Artiemjew, "On a Hybridization of Deep Learning and Rough Set Based Granular Computing," *Algorithms*, vol. 13, no. 3, p. 63, 2020. [Online]. Available: doi.org/10.3390/a13030063
- [106] P. Artiemjew and K. Ropiak, "A Novel Ensemble Model – The Random Granular Reflections," *Fundamenta Informaticae*, vol. 179, no. 2, pp. 183–203, 2021. [Online]. Available: doi.org/10.3233/FI-2021-2020
- [107] G. Toppin, J. Borkowski, D. Ślęzak, S. Shi, P. Synak, J. Wróblewski, T. J. Wongkee, and G. Charalabopoulos, "System and Method for Granular Scalability in Analytical Data Processing," US Patent Application 20150088807, 2014.
- [108] M. Przyborski, T. Tajmajer, Ł. Grad, A. Janusz, P. Biczky, and D. Ślęzak, "Toward Machine Learning on Granulated Data – a Case of Compact Autoencoder-based Representations of Satellite Images," in *IEEE International Conference on Big Data (IEEE BigData 2018)*, Seattle, WA, USA, December 10-13, 2018, N. Abe, H. Liu, C. Pu, X. Hu, N. K. Ahmed, M. Qiao, Y. Song, D. Kossman, B. Liu, K. Lee, J. Tang, J. He, and J. S. Saltz, Eds. IEEE, 2018, pp. 2657–2662. [Online]. Available: doi.org/10.1109/BigData.2018.8622562
- [109] G. D. Tré, T. Boeckling, Y. Timmerman, and S. Zadrozny, "Handling Veracity of Nominal Data in Big Data: A Multipolar Approach," in *Flexible Query Answering Systems – 13th International Conference, FQAS 2019, Amantea, Italy, July 2-5, 2019, Proceedings*, ser. Lecture Notes in Computer Science, A. Cuzzocrea, S. Greco, H. L. Larsen, D. Saccà, T. Andreassen, and H. Christiansen, Eds., vol. 11529. Springer, 2019, pp. 317–328. [Online]. Available: doi.org/10.1007/978-3-030-27629-4_29
- [110] R. S. Geiger, D. Cope, J. Ip, M. Lotosh, A. Shah, J. Weng, and R. Tang, "'Garbage In, Garbage Out' Revisited: What Do Machine Learning Application Papers Report about Human-Labeled Training Data?" *Quantitative Science Studies*, vol. 2, no. 3, pp. 795–827, 2021. [Online]. Available: doi.org/10.1162/qss_a_00144
- [111] M. Kassen, *Open Data Governance and Its Actors – Theory and Practice*, ser. Studies in National Governance and Emerging Technologies. Palgrave Macmillan, 2022. [Online]. Available: doi.org/10.1007/978-3-030-92065-4
- [112] D. Plotkin, *Data Stewardship: An Actionable Guide to Effective Data Management and Data Governance, 2nd Edition*. Academic Press, 2020.
- [113] A. Der Kiureghian and O. Ditlevsen, "Aleatory or Epistemic? Does It Matter?" *Structural Safety*, vol. 31, no. 2, pp. 105–112, 2009. [Online]. Available: sciencedirect.com/science/article/pii/S0167473008000556
- [114] D. Ślęzak and A. Chądzyńska-Krasowska, "Approximate Decision Tree Induction over Approximately Engineered Data Features," in *Rough Sets – International Joint Conference, IJCRS 2020, Havana, Cuba, June 29 – July 3, 2020, Proceedings*, ser. Lecture Notes in Computer Science, R. Bello, D. Miao, R. Falcon, M. Nakata, A. Rosete, and D. Ciucci, Eds., vol. 12179. Springer, 2020, pp. 376–384. [Online]. Available: doi.org/10.1007/978-3-030-52705-1_28
- [115] B. Settles, "Active Learning Literature Survey," University of Wisconsin–Madison, Computer Sciences Technical Report 1648, 2009.
- [116] S. Dutta and A. Skowron, "Interactive Granular Computing Connecting Abstract and Physical Worlds: An Example," in *Proceedings of the 29th International Workshop on Concurrency, Specification and Programming (CS&P 2021), Berlin, Germany, September 27-28, 2021*, ser. CEUR Workshop Proceedings, H. Schlingloff and T. Vogel, Eds., vol. 2951. CEUR-WS.org, 2021, pp. 46–59. [Online]. Available: ceur-ws.org/Vol-2951/paper18.pdf
- [117] L. A. Zadeh, Ed., *Computing with Words: Principal Concepts and Ideas*, ser. Studies in Fuzziness and Soft Computing. Springer, 2012, vol. 277. [Online]. Available: doi.org/10.1007/978-3-642-27473-2
- [118] J. Pearl, "Causal Inference in Statistics: An Overview," *Statistics Surveys*, vol. 3, pp. 96–146, 2009. [Online]. Available: doi.org/10.1214/09-SS057
- [119] A. Janusz, D. Kałuża, A. Chądzyńska-Krasowska, B. Konarski, J. Holland, and D. Ślęzak, "IEEE BigData 2019 Cup: Suspicious Network Event Recognition," in *2019 IEEE International Conference on Big Data (IEEE BigData), Los Angeles, CA, USA, December 9-12, 2019*, C. K. Baru, J. Huan, L. Khan, X. Hu, R. Ak, Y. Tian, R. S. Barga, C. Zaniolo, K. Lee, and Y. F. Ye, Eds. IEEE, 2019, pp. 5881–5887. [Online]. Available: doi.org/10.1109/BigData47090.2019.9005668
- [120] M. Matraszek, A. Janusz, M. Świechowski, and D. Ślęzak, "Predicting victories in video games – IEEE bigdata 2021 cup report," in *2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, December 15-18, 2021*, Y. Chen, H. Ludwig, Y. Tu, U. M. Fayyad, X. Zhu, X. Hu, S. Byna, X. Liu, J. Zhang, S. Pan, V. Papalexakis, J. Wang, A. Cuzzocrea, and C. Ordonez, Eds. IEEE, 2021, pp. 5664–5671. [Online]. Available: doi.org/10.1109/BigData52589.2021.9671650
- [121] F. P. Brooks, Jr., *The Mythical Man-Month: Essays on Software Engineering, Anniversary Edition*. Addison-Wesley, 1995.
- [122] S. Dutta, A. Skowron, and M. K. Chakraborty, "Information Flow in Logic for Distributed Systems: Extending Graded Consequence," *Information Sciences*, vol. 491, pp. 232–250, 2019. [Online]. Available: doi.org/10.1016/j.ins.2019.03.057
- [123] A. Skowron, A. Jankowski, and P. Wasilewski, "Interactive Computational Systems: Rough Granular Approach," in *Proceedings of the 21th International Workshop on Concurrency, Specification and Programming, Berlin, Germany, September 26-28, 2012*, ser. CEUR Workshop Proceedings, L. Popova-Zeugmann, Ed., vol. 928. CEUR-WS.org, 2012, pp. 358–369. [Online]. Available: ceur-ws.org/Vol-928/0358.pdf
- [124] C. Savaglio, M. Ganzha, M. Paprzycki, C. Badica, M. Ivanovic, and G. Fortino, "Agent-based Internet of Things: State-of-the-Art and Research Challenges," *Future Generation Computer Systems*, vol. 102, pp. 1038–1053, 2020. [Online]. Available: doi.org/10.1016/j.future.2019.09.016

Modern C++ in the era of new technologies and challenges—why and how to teach modern C++?

Bogusław Cyganek

¹AGH University of Science and Technology, Poland
 Al. Mickiewicza 30, 30-059 Kraków, Poland

²Academic Computer Center Cyfronet AGH
 Ul. Nawojki 11, 30-950 Kraków, Poland
 cyganek@agh.edu.pl

Abstract—Computers are one of the most important inventions in human history, and computer languages enable human-computer communication. Undoubtedly, C++ is one of the most important and influential in this group. Nevertheless, new technologies and related industry challenges place high demands on C++ and foster the development of new computer languages that meet new needs. For this reason, and thanks to the dynamically operating ISO standardization group, C++ is constantly updated while maintaining its backward compatibility. However, all this complicates and hinders not only the teaching of beginners but also the use by professionals. In this article, we briefly discuss the goals as well as proposed methodologies and techniques for teaching contemporary C++ in the age of new technologies and challenges.

Index Terms—C++, modern technologies, compilers, teaching programming, computer science curricula

I. INTRODUCTION

C++ is a multi-paradigm, imperative, procedural, functional, object-oriented, generic, and modular language invented in early 1980s and further developed by Bjarne Stroustrup [5][10] [29]. Since 1991 standardization of C++ is supported by the ANSI International Organization for Standardization (ISO), with the latest standard version published in December 2020 [21]. With performance and efficiency in mind, C++ extends and is compatible with the C programming language, while to incorporate the object-oriented and abstraction mechanisms it draws from Simula. This hybrid approach has proven extremely useful over the years, especially in such domains as systems programming, embedded systems, resource constrained platforms, large computing and simulation libraries, machine learning & artificial intelligence (ML/AI) and many others.

Nevertheless, there are industries such as web applications that favor the development of other languages as well. While the popularity rankings of programming languages are in some ways superficial and may be misleading, they provide some

insight into future trends in the IT industry and can help students decide which language they want to learn. From these the TIOBE Programming Community index shows the popularity of programming languages based on 25 search engines [16]. At its top are Python, C, Java, and C++, which together are well ahead of the others, as shown in Fig. 1. In the last two years, Python and C have swapped between 1st and 2nd places in the ranking. Also C++ is gaining in popularity and tends to surpass Java.

Aug 2022	Aug 2021	Change	Programming Language	Ratings	Change
1	2	▲	 Python	15.42%	+3.56%
2	1	▼	 C	14.59%	+2.03%
3	3		 Java	12.40%	+1.96%
4	4		 C++	10.17%	+2.81%
5	5		 C#	5.59%	+0.45%

Fig. 1 An excerpt from the TIOBE list of the top-ranked programming languages in 2022 (from [16]).

On the other hand, in the Popularity of Programming Language Index (PYPL), which shows how often language tutorials are searched on Google, C/C++ are ranked 5th together (Aug. 2022) [17]. However, neither of the above indexes is about the best programming language or the language in which most lines of code have been written.

A detailed analysis of various languages and their applications is far from the scope of this paper, nevertheless we can observe that while scripting tasks surely fall into the realm of Python, and web development for Java, then vast majority of high performance applications falls into the domain of C/C++.

The latter are also the only languages in this group that compile their code directly into the machine language. Although, there are contenders such as Rust and the recent Carbon [11][12], C++ endowed with hundreds of libraries, tools, and many years of experience, and a superset of C in a sense, *is and will probably be the most important and productive language today*, especially for large and performance demanding systems. Therefore, C++ is surely worth learning. However, the more extensive the specification of modern C++ becomes, the more critical the requirement to properly teach modern C++ to new generations of programmers becomes. In this article, we tackle this issue in an attempt to shed more light on why and how to teach modern C++.

There are relatively large Internet resources [30][10][13] and literature [27][20][23] about C++ and its features. However, when it comes to teaching modern C++, the situation is not bright. There are only few online presentations [2][3][4], web services and books to recommend [28][29]. Nevertheless, even these are a bit dated considering new C++17 and C++20 standards. To fill this gap the new book was written, *Introduction to Programming with C++ for Engineers*, which was published by Wiley-IEEE Press in 2021 [6]. It contains teaching materials, from elementary to advanced level, intended for the three-semester study cycle. Based on this, this article provides an overview of methodology and techniques for teaching modern C++.

It is worth mentioning that the problem of teaching and disseminating knowledge about modern C++ also found wide interest in the language committee. In this context, the SG20 group arose, whose aim is to prepare and provide guidelines for content to be covered by C++ courses [31]. Their main document is a resource for instructors to assist in the preparation of C++ courses in a variety of environments, including universities, colleges, and industry.

The rest of the paper is organized as follows. An overview of the methodologies and techniques of teaching the C++ language is presented in Section II. It is organized into four subsections. Section III provides scenarios for the different levels of C++ learning. Section IV discusses the role of good examples in the teaching process. Section V deals with the issue of teaching for real life challenges. The paper ends with conclusions in Section VI.

II. AN OVERVIEW OF TEACHING METHODOLOGIES & TECHNIQUES

In this section the basic methodologies and techniques for teaching modern C++ are outlined. The main issue is to list the most important steps in class preparation and to focus on the most important factors.

A. Preparing for Teaching

First, there are some key factors to consider before starting your class. The following is a list of them:

- Get to know your students – what are their backgrounds, what are their motivations, whether they are kids or students of electronics and telecommunication, computer science students, or students of non-technical faculties (biology, humanistic, etc.); or professionals

who want to expand their skills in modern C++? What have they already learned, math, python, basics of computer science?

- Organize your classes well – individual or group work (some activities such as lectures can be for a group, but some – such as tutorial – should be individual), group sizes, etc. Have a plan but actively respond to students' progress and expectations, have close contact and react actively, similarly to the *agile* methodology for software development. But also control the attitude and experience of other fellow teachers in the group (in many universities often the lecturer and laboratory teacher are different people).
- Plan your time – how many hours for a lecture, for a lab, for joint work, and for an individual project. Consider time for individual consultations.
- Organize the class work well – consider exercises for personal work as well as projects for *team work*.
- Teaching materials – students have access to various sources, but they rely on your opinions, therefore the correct selection of book(s), internet materials, video(s), etc. is very important.

Certainly, these are only propositions based on many years of our observations and conducted classes. However, for different groups and teachers, the list and importance of each factor may differ.

B. Choice of the Vital Language Features – the 20/80 Rule

What works well in our 25 years teaching experience is getting the right preparation and then focusing on the most important and productive features of the language at the given teaching stage that allow students to quickly comprehend and become proficient in basic programming techniques. As a result, it allows the students to create useful and well-organized programs as quickly as possible. The choice of features can be arbitrary, but is best if these are based on the experience of the teacher(s). As we have noticed, for this purpose *the Pareto 20/80 principle* is worth considering [14][6].

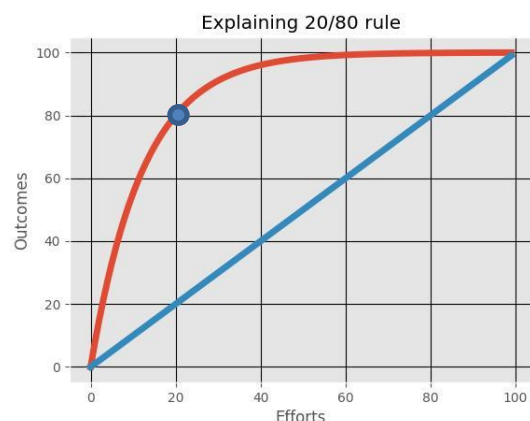


Fig. 2 The 20/80 rule states that many activities are not evenly distributed and some contribute more than others. This idea can be used to prepare 20% of the most important C++ features for the students' classes.

The 20/80 paradigm, known also as the Pareto rule or law of the vital few, states that in many life situations, 20% of causes is responsible for roughly 80% of consequences, or results, as shown in Fig. 2. This heuristic observation was probably first noticed by the mentioned Italian economist Vilfredo Pareto, who noted that at that time 80% of the land in Italy was owned by 20% of the population. Interestingly, this can also be observed in computers, which is usually manifested that 20% of bugs contribute 80% of crashes, or that 80% of the CPU time is spent on 20% of the code, etc.

Hence, the idea is to use this rule to prepare the most important 20% of features to be taught in the beginning. This approach can result in much better productivity and allows students to faster reach the level of solid understanding of basic programming constructs and techniques, compared e.g. to the linear approach, as shown in Fig. 2.

C. The Spiral Development Model

As originally proposed by Boehm [1][15], the spiral model of software development is associated with iteratively repeated processes while managing risk for its active reduction, as shown in Fig. 3.

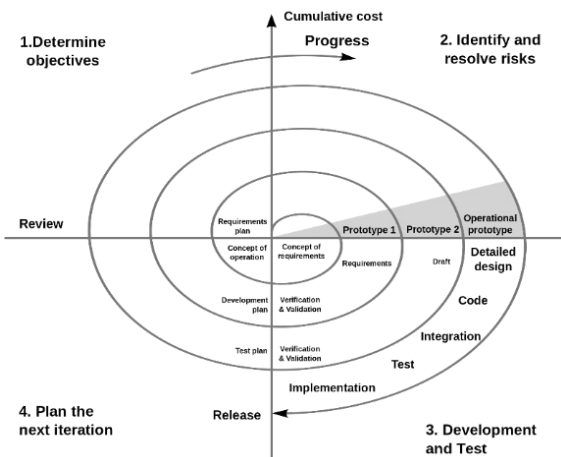


Fig. 3 The spiral model, originally proposed by Boehm for development of software, can be also applied to the C++ teaching process (from [1]).

However, it appears that the C++ teaching process is in many ways similar to this spiral model. Namely, many topics are, and should be, repeated with wider scope and level of details. Risk management in this case means strict control of the students' absorption of previous material before presenting an advanced version of a topic. This also applies to the gradual increase in the complexity of student developed projects.

III. EXEMPLARY TEACHING SCENARIOS

Some scenarios for teaching modern C++ are discussed here. Assuming classes organized in the semester periods (14/15 weeks per semester), and organized in the form of a lecture per week, a laboratory per week for the half of the semester, as well as the student's own project and consultations

for the remaining part of the semester, the following scenarios are presented for the entire three-semester C++ teaching cycle:

1. Introduction to programming with C++ (the beginners program), followed by basics of C++.
2. Object-oriented design & programming with C++, followed by advanced memory management.
3. Advanced C++ concepts, followed by basics of parallel programming.

A possible organization of classes in the form of a state diagram is shown in Fig. 4. Each of the three semesters consists of *two building states* – the idea here is that the second state in a semester is optional, i.e. it is undertaken if there is enough time and the group have achieved good results in the first state.

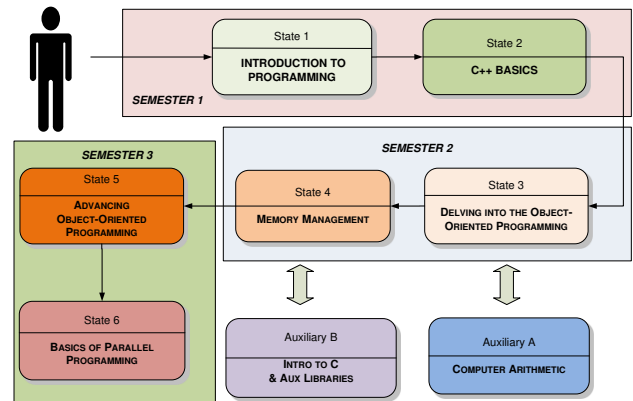


Fig. 4 Possible C++ teaching scenarios organized for the three semesters. Each semester consists of two stages – compulsory and optional, which is carried out if there is enough time and the group has demonstrated significant progress in learning. Also visible are auxiliary topics “A” for computer arithmetic and “B” for low-level features, such as the C programming language and introduction to additional libraries.

These are supplemented with the auxiliary states, which can be included to the program of teaching semesters depending on the needs and progress of the students. Supporting topics include: “A” computer arithmetic and “B” introduction to programming in C and a supplemental introduction to using libraries such as QT, FLTK, OpenGL, OpenMP, OpenCV, etc., depending on the students' needs and the profile of their faculties.

However, before providing some more concrete lists of features for each teaching scenario, let's highlight the following issues and hints that should be considered:

- Well define the main goals of the classes.
- For each semester well define a minimal set of C++ features to be acquired by students; for this purpose the 20/80 rule can be applied.
- Throughout the term: stick to the developed teaching plan (the curriculum) but actively respond to students' progress – this resembles the *agile* concept, applied to the teaching process.

Not surprisingly, the aforementioned teaching scenarios follow chapter layout in the book [6]. The more detailed teaching scenarios are outlined in the following subsections.

A. Scenario for Beginners

Following the plan outlined in Fig. 4 let's analyze a possible minimal set of C++ features. This can be defined as follows.

1. Introduction to the computer API and the basic C++ development tools (editor, compiler, linker, IDE, etc.).
2. The `main` function.
3. Minimal libraries (`#include`), using directive.
4. Printing texts `std::cout`.
5. Defining and *initializing* variables: `int` and `double` (explain the difference).
6. Entering values to the variables `std::cin`.
7. Conditional statement `if` and how to provide a logical condition.
8. `std::vector`
9. `std::string`
10. The loop: "classical" `for` and "range" `for`

Especially for beginners it is important to provide diagrams and ready recipes for project organization, tools, build process, particularly compiling and dealing with compilation errors as well as basics of debugging. Active support from the teacher is essential at this stage. A possible diagram of an exemplary program in the single `main` function is shown in Fig. 5.

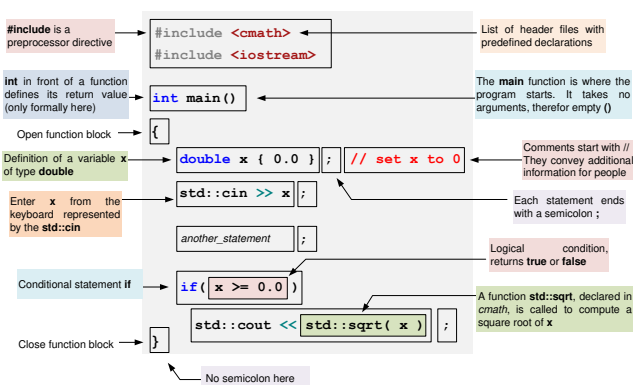


Fig. 5 A diagram showing basic constructions of a C++ program for absolute beginners (from [6]). Many projects for beginners can be written on the canvas of a single `main` function.

Although very simple, the code based on a single `main` function can be used successfully in many beginner projects.

B. Scenario for C++ Basics

A possible scenario for teaching the basics of C++ may include the following topics.

1. The most common built-in data types, their applications and initialization.
2. Code debugging techniques.
3. Basic members and applications of `std::vector`.
4. A matrix as a vector of vectors.

5. Basics of `std::string`.
6. `auto` and when to use it.
7. Common standard algorithms (`std::copy`, `std::find`, `std::generate`, `std::accumulate`, etc.).
8. Structures as data containers with `struct`.
9. References.
10. Statements (the role of braces).
11. Functions (argument passing, recursive, lambdas).
12. Intro to (separate) classes: `struct` vs `class` + constructor and member functions.
13. Basic of software testing.
14. Operators.

The beginners and basics scenarios constitute teaching material for the 1st semester (Fig. 4).

C. Scenario for Object-Oriented Programming in C++

A natural follow up is introduction to the OOP domain with C++. At this stage a possible list of topics can look as follows.

1. Intro to OOP.
2. Anatomy of a class (e.g. extended matrix class).
3. Right references.
4. Classes with all special functions – move semantics explained (e.g. extending matrices into tensors).
5. Templates and generic programming (functions, classes, member templates).
6. Virtual mechanism.
7. Some design patterns (e.g. wrapper, handle-bode/bridge, proxy).
8. Memory management (RAII, `std::unique_ptr`, `std::shared_ptr`, `std::weak_ptr`).

The virtual mechanisms and polymorphism should be shown on class hierarchies. However, the more complex are postponed to the advanced level, as discussed in the next section. Nevertheless, the introduction to the OOP and topics of memory management constitute material for the 2nd semester.

D. Teaching Advanced Topics of C++

After completing the OOP and memory management parts of the course, the last stage can be coined "advanced C++". However, it is not less difficult to define what actually should be included and how to teach such advanced concepts. Nevertheless, a possible list of topics can be formed as follows.

1. Designing class hierarchies.
2. C++ filesystem.
3. Forward/universal references.
4. Regular expressions (`std::regex`).
5. Graphical user interface (various libraries, QT, MFC, FLTK, ...).
6. System clock and time measurement (`std::chrono`).
7. Intro to functional programming and the `std::ranges`.
8. Intro to expression parsing – the interpreter DP, building the syntax trees, composite DP, visitor DP.
9. State machine pattern.
10. Advanced generic programming techniques.

11. Basics of parallel programming (`std::par`, `std::thread`, OpenMP).

Certainly, all the aforementioned teaching scenarios are just simple propositions [6]. They can be easily adjusted to best suit the level and needs of the students as already discussed.

IV. THE IMPECCABLE ROLE OF GOOD EXAMPLES

As it is not possible to teach swimming without entering the water, it is also not possible to teach programming without writing code and developing projects by students. Hence, the role of good code examples cannot be overestimated. However, let's consider what factors should be taken into account when preparing code examples for teaching. A good example of code should be:

- comprehensible,
- not too long,
- touch on 'real' problems,
- can serve as a starting point for student's project; i.e. it can be used in an incremental development and the spiral model in action.

The examples are very important because not only do they illustrate some code concepts, but they provide "thought patterns" that the developer evokes and then modifies while working on a similar problem.

There are different ways students can use the code examples, for instance:

- Compile, make it run, debug to observe the results.
- Re-type an example and make it run.

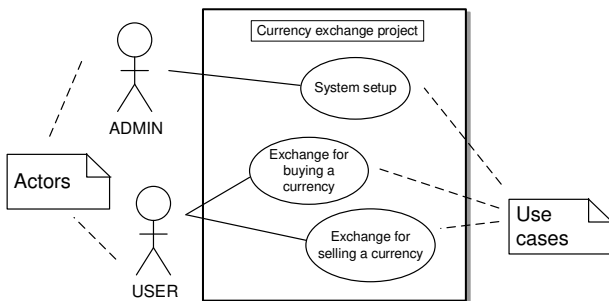


Fig. 6 Teaching software development is about more than showing C++ code. The entire process should be presented, starting e.g. with the UML diagrams and the complete development process, from design, till code unity tests and deployment. Here is a UML use case diagram of the currency exchange exemplary project (adopted from [6]).

- Don't use the example – create your own version, instead; then compare.
- Use as a 'startup' for your own project/example.

It is also important for educators to realize that teaching programming should encompass *the deeper context of software development*. This means that instead of showing only parts of C++, a teacher should present the entire development process,

starting with problem analysis, resulting UML diagrams such as in Fig. 6, software design, code unity tests, deployment, etc.

However, we'd like to emphasize that not less important is the process of consultations between the student and her/his teacher. Such a code refinement process enables comprehension of proper coding techniques.

V. TEACHING FOR THE REAL-LIFE CHALLENGES

The world did not start today, and there are millions of lines of the legacy code that still need to be maintained, deployed, refactored or extended. Therefore, teachers should ask themselves questions such as how to prepare students to cope with the existing code, as well as how to respond to the expectations of employers. The problem is even deeper – there is an ongoing discussion on mismatch between computer science curricula and industry needs [22][24][25][26][32], which generally is out of the scope of this paper. However, when confining this to teaching modern C++ the teachers should be aware of the *what* to address and *how* to go about, as well as of the *short time span* issues.

Returning to the task of teaching modern C++ in the light of the legacy code, we need to introduce the lower-level constructions of C++ (such as pointers, unions, raw memory/string operations, etc.), sometimes also at least the basics of C. Certainly, the set of these low-level features depends on the needs of the students. However, the most important are the proper moments in the teaching scenarios when these low-level features are taught. That is, low-level features should be introduced *after* instilling good programming habits in using *modern* features of C++, not the other way around.

The low-level features will be inevitable when preparing students to work with operating systems such as Linux or FreeRTOS, or to use many common libraries, such as OpenMP, OpenCV, OpenGL, QT or games with Unreal Engine, etc. But even to explain at some point what is `int main(int argc, char ** argv)` the teacher has to face how to introduce the pointers. So the question is not "if" but "when".

VI. CONCLUSIONS

The paper contains a short overview of the challenges related to promoting the interest in modern C++, as well as in teaching modern C++ to various groups of learners in the era of new technological challenges. The subject is very wide and we only scratched the surface.

Every three years, the C++ standardizing committee, supported by thousands of enthusiasts, publishes a new specification for the language. Thanks to this, it gained dozens of new features, which makes it very efficient and effective. However, such a dynamically changing environment also raises some problems especially in terms of the *stability* of language features as well as their presentation and acceptance from the world C++ programming community.

This also makes teaching modern C++ a real challenge. Accordingly, this paper introduces some teaching methodologies and techniques, such as the 20/80 principle, as well as some modifications to the software development

methodologies, such as the spiral model and agile approach, which were brought into the teaching domain. Further on, the overviews of some teaching scenarios for different groups of learners were provided. More detailed versions are available in the book *Introduction to Programming with C++ for Engineers* [6], whereas the code examples and additional materials are available from the Internet [8][9].

In the end, the following list summarizes the main postulates and guidelines for effective teaching of modern C++:

- As C++ programmers we are all students and many of us are, or become, teachers.
- When teaching, get to know your students, get to know their needs and wishes, then organize the classes well.
- At each stage think about selecting the appropriate C++ features for teaching, keeping in mind *the 20/80 rule*.
- Provide good and practical examples! They constitute “mental patterns” for your students.
- Keep things simple and in right order – but be *agile* and actively react to students’ progress.
- Teach in a repeated way, gradually introduce more advanced concepts and techniques, raising the level – apply the *spiral* development model.
- Teach not only C++ itself, but C++ across *the entire computer science framework*: show steps of software design, UML, data structures, algorithms, design patterns, development tools, software testing, etc.
- Do not forget about the programmers/companies reality, the legacy code, etc. Provide information on lower-lever features, teach basics of C if necessary, but at the right time.
- Pay attention to the self-education, always improve the skills of not only your students, but also yourself and your team, watch/participate in lectures, conferences, read books, etc.

With the increasing demands on high performance systems, the embedded world moving from C to C++, the revolution in big data, parallel computations, etc., undeniably C++ is, and probably will be, the most powerful modern computing language for many years to come. Therefore, there is no doubt that *C++ should be taught to a wide range of students*, especially of technical faculties. However, this should be well organized as *one of the important topics* in wisely prepared computer science curricula, containing *a combination of various programming languages*, that respond to the needs of industry in the era of new technologies and challenges.

As a concluding remark let's remember that: *The quality of the software of the future depends on the quality of education today.*

ACKNOWLEDGMENT

The author expresses his gratitude to Prof. Dominik Ślęzak for his invitation and encouragement to write this paper.

The author also commends Wiley-IEEE Press for the 2021 Wiley-IEEE Press Professional Book Award for the book *Introduction to Programming with C++ for Engineers* (<https://ieeepress.ieee.org/wiley-ieee-press-awards/>) and for financial support in participation in FedCSIS'22.

REFERENCES

- [1] Boehm, B: Spiral Development: Experience, Principles, and Refinements. Special Report. Software Engineering Institute, 2000.
- [2] CppCon 2017: Bjarne Stroustrup “Learning and Teaching Modern C++” – YouTube <https://www.youtube.com/watch?v=fX2W3nNjJI0>
- [3] CppCon 2015: Kate Gregory “Stop Teaching C” – YouTube <https://www.youtube.com/watch?v=YnWhqhNdYyk>
- [4] CppCon 2018: Christopher Di Bella “How to Teach C++ and Influence a Generation”—YouTube <https://www.youtube.com/watch?v=3AkPd9Nt2Aw>
- [5] C++ Wikipedia: <https://en.wikipedia.org/wiki/C%2B%2B>
- [6] Cyganek B.: Introduction to Programming with C++ for Engineers. Wiley-IEEE Press, 2021.
- [7] Gurcan, F., Kose, C.: Analysis of software engineering industry needs and trends: implications for education. International Journal of Engineering Education, Vol. 33, pp. 1361-1368, 2017.
- [8] <https://home.agh.edu.pl/~cyganek/BookCpp.htm>
- [9] <https://github.com/BogCyg/BookCpp>
- [10] <https://cppreference.com>
- [11] <https://thenewstack.io/google-launches-carbon-an-experimental-replacement-for-c/>
- [12] <https://9to5google.com/2022/07/19/carbon-programming-language-google-cpp/>
- [13] <https://stackoverflow.org>
- [14] https://en.wikipedia.org/wiki/Pareto_principle
- [15] https://en.wikipedia.org/wiki/Spiral_model
- [16] <https://www.tiobe.com/tiobe-index/>
- [17] <https://pypl.github.io/PYPL.html>
- [18] <https://www.devjobsscanner.com/blog/top-8-most-demanded-languages-in-2022/>
- [19] History of C++ <https://en.cppreference.com/w/cpp/language/history>
- [20] Josuttis N.: C++17 - The Complete Guide: First Edition, 2019.
- [21] JTC1/SC22/WG21 - The C++ Standards Committee – ISO C++, 2022. <https://www.open-std.org/jtc1/sc22/wg21/>
- [22] Lawlis P.K., Adams K.A.: Computing Curricula vs. Industry Needs: A Mismatch. Proc. 9th Annual ASEET Symposium, pp. 5-19, 1995.
- [23] Meyers S.: Effective Modern C++: 42 Specific Ways to Improve Your Use of C++11 and C++14, 2014.
- [24] Moreno A.M., Sanchez-Segura M-I, Medina-Dominguez F., Carvajal L.: Balancing software engineering education and industrial needs, Journal of Systems and Software, Volume 85, Issue 7, pp 1607-1620, 2012.
- [25] Oguz, D., Oguz, K.: Perspectives on the Gap Between the Software Industry and the Software Engineering Education. IEEE Access, Vol. 7, pp. 117527-117543, 2019.
- [26] Paprzycki M., Zalewski J.: CS II: An Applied Software Engineering Project. The Journal of Computing in Small Colleges, Vol. 12, No., pp. 47-52, 2, 1996.
- [27] Stroustrup B.: The C++ Programming Language, Addison-Wesley, 2013.
- [28] Stroustrup B.: Programming: Principles and Practice Using C++, 2nd Ed., Addison-Wesley, 2014.
- [29] Stroustrup B.: A Tour of C++, Addison-Wesley, 2018.
- [30] Stroustrup B., Sutter H.: C++ Core Guidelines, 2022. <https://isocpp.github.io/CppCoreGuidelines/CppCoreGuidelines>
- [31] SG20 (ISO C++ Study Group on Education): Guidelines for Teaching C+++, 2022. <https://cplusplus.github.io/SG20/latest/>
- [32] Waks S., Frank M.: Engineering Curriculum versus Industry Needs – A Case Study. IEEE Tr. on Education, Vol. 43, No. 3, pp. 349-352, 2000.

17th International Symposium Advances in Artificial Intelligence and Applications

THIS track is a continuation of international AAIA symposiums, which have been held since 2006. It aims at establishing the synergy between technical sessions, which encompass wide range of aspects of AI. With its longest-tradition threads, such as WCO focusing on Computational Optimization, it is also open to new initiatives categorized with respect to both, the emerging AI-related methodologies and practical usage areas. Nowadays, AI is usually perceived as closely related to the data, therefore, this track's scope includes the elements of Machine Learning, Data Quality, Big Data, etc. However, the realm of AI is far richer and our ultimate goal is to show relationships between all of its subareas, emphasizing a cross-disciplinary nature of the research branches such as XAI, HCI, and many others.

AAIA'22 brings together scientists and practitioners to discuss their latest results and ideas in all areas of Artificial Intelligence. We hope that successful applications presented at AAIA'22 will be of interest to researchers who want to know about both theoretical advances and latest applied developments in AI.

TOPICS

Papers related to theories, methodologies, and applications in science and technology in the field of AI are especially solicited. Topics covering industrial applications and academic research are included, but not limited to:

- Decision Support
- Machine Learning
- Fuzzy Sets and Soft Computing
- Rough Sets and Approximate Reasoning
- Data Mining and Knowledge Discovery
- Data Modeling and Feature Engineering
- Data Integration and Information Fusion
- Hybrid and Hierarchical Intelligent Systems
- Neural Networks and Deep Learning
- Reinforcement Learning
- Bayesian Networks and Bayesian Reasoning
- Case-based Reasoning and Similarity
- Web Mining and Social Networks
- Business Intelligence and Online Analytics
- Robotics and Cyber-Physical Systems
- AI-centered Systems and Large-Scale Applications
- AI for Combinatorial Games, Video Games and Serious Games
- Evolutionary Algorithms and Evolutionary Computation
- Artificial Intelligence for Next-Generation Diagnostic Imaging (1st Workshop AI4NextGenDI'22)

- Artificial Intelligence for Patient Empowerment with Sensor Systems (1st Workshop AI4Empowerment'22)
- Artificial Intelligence in Machine Vision and Graphics (4th Workshop AIMaViG'22)
- Intelligent Ambient Assisted Living Systems (1st Workshop IntelligentAAL'22)
- Personalization and Recommender Systems (1st Workshop PeRS'22)
- Rough Sets: Theory and Applications (4th International Symposium RSTA'22)
- Computational Optimization (15th Workshop WCO'22)

TRACK CHAIRS

- **Zdravevski, Eftim**, Ss. Cyril and Methodius University, Macedonia
- **Szczuka, Marcin**, University of Warsaw, Poland
- **Matwin, Stan**, Dalhousie University, Canada

PROGRAM CHAIRS

- **Corizzo, Roberto**, American University, USA
- **Sosnowski, Łukasz**, Systems Research Institute, Polish Academy of Sciences, Poland
- **Świechowski, Maciej**, QED Software, Poland

PROGRAM COMMITTEE

- **Azad, Mohammad**, Jouf University, Saudi Arabia
- **Bellinger, Colin**, National Research Council of Canada – Ottawa, Canada
- **Bianchini, Monica**, Dipartimento di Ingegneria dell'Informazione, Università di Siena, Italy
- **Boukouvalas, Zois**, American University – Washington DC, USA
- **Calpe Maravilla, Javier**, University of Valencia, Spain
- **Chelly, Zaineb**, Université Paris-Saclay, UVSQ, DAVID, France
- **Colantonio, Sara**, ISTI-CNR, Italy
- **Corizzo, Roberto**, American University, USA
- **Cyganek, Bogusław**, AGH University of Science and Technology, Poland
- **Dey, Lipika**, TCS Innovation Lab Delhi, India
- **Durães, Dalila**, Universidade do Minho, Portugal
- **Filipe, Vitor**, UTAD, Portugal
- **Girardi, Rosario**, UFMA, Brasil
- **Goleva, Rossitza**, New Bulgarian University, Bulgaria
- **Hullam, Gabor**, Budapest University of Technology and Economics, Hungary

- **Hussain, Shahid**, Institute of Business Administration, Pakistan
- **Jakovljevic, Niksa**, University of Novi Sad, Faculty of Technical Sciences, Bulgaria
- **Jaromczyk, Jerzy W.**, University of Kentucky, USA
- **Kaczmarek, Katarzyna**, Instytut Badań Systemowych Polskiej Akademii Nauk, Poland
- **Kasprzak, Włodzimierz**, Warsaw University of Technology, Poland
- **Kwaśnicka, Halina**, Wrocław University of Technology, Poland
- **Laskov, Lasko**, New Bulgarian University, Bulgaria
- **Lerga, Jonatan**, Rijeka Technical University, Croatia
- **Lingras, Pawan**, Saint Mary's University, Canada
- **Ljubić, Sandi**, University of Rijeka, Faculty of Engineering, Croatia
- **Matwin, Stan**, Dalhousie University, Canada
- **Meneses, Claudio**, Universidad Católica del Norte, Chile
- **Mignone, Paolo**, Università degli studi di Bari, Italy
- **Mihajlov, Martin**, Jozef Stefan Institute, Slovenia
- **Moore, Neil**, University of Kentucky, Department of Computer Science, USA
- **Moshkov, Mikhail**, King Abdullah University of Science and Technology, Saudi Arabia
- **Mozgovoy, Maxim**, University of Aizu, Japan
- **Muñoz, Andrés**, Universidad de Cádiz, Spain
- **Myszkowski, Paweł**, Wrocław University of Science and Technology, Poland
- **Noguera, Manuel**, University of Granada, Spain
- **Pataricza, András**, Budapest University of Technology and Economics, Hungary
- **Peters, Georg**, Munich University of Applied Sciences & Australian Catholic University, Germany
- **Petrovska, Biserka**, Goce Delcev University, North Macedonia
- **Pires, Ivan Miguel**, Instituto de Telecomunicações, Universidade da Beira Interior, and Universidade de Trás-os-Montes e Alto Douro, Portugal
- **Po, Laura**, Università di Modena e Reggio Emilia, Italy
- **Porta, Marco**, University of Pavia, Italy
- **Przybyła-Kasperek, Małgorzata**, Uniwersytet Śląski, Poland
- **Raghavan, Vijay**, University of Louisiana at Lafayette, USA
- **Ramanna, Sheela**, Department of Applied Computer Science, University of Winnipeg, Canada
- **Rauch, Jan**, University of Economics, Prague, Czech Republic
- **Schaefer, Gerald**, Loughborough University, United Kingdom
- **Sosnowski, Łukasz**, Systems Research Institute, Polish Academy of Sciences, Poland
- **Spinsante, Susanna**, Università Politecnica delle Marche, Italy
- **Stańczyk, Urszula**, Silesian University of Technology, Poland
- **Stoian, Catalin**, University of Craiova, Romania
- **Stojanov, Riste**, Faculty of Computer Science and Engineering, Skopje
- **Subbotin, Sergey**, Professor, Zaporozhye National Technical University, Ukraine
- **Szczuka, Marcin**, University of Warsaw, Poland
- **Szczęch, Izabela**, Poznan University of Technology, Poland
- **Trajkovic, Vladimir**, Ss. Cyril and Methodius University, Faculty of Computer Science and Engineering, North Macedonia
- **Turukalo, Tatjana Loncar**, FTN, University of Novi Sad, Bulgaria
- **Unland, Rainer**, University of Duisburg-Essen, ICB, Germany
- **Verstraete, Jörg**, Instytut Badań Systemowych Polskiej Akademii Nauk, Poland
- **Weber, Richard**, Universidad de Chile
- **Zakrzewska, Danuta**, Institute of Information Technology, Technical University of Lodz, Poland
- **Zdravevski, Eftim**, Ss. Cyril and Methodius University, North Macedonia
- **Zielosko, Beata**, University of Silesia, Institute of Computer Science, Poland
- **Świechowski, Maciej**, QED Software, Poland
- **Żurek, Dominik**, AGH, Poland

A novel link prediction approach on clinical knowledge graphs utilising graph structures

Jens Dörpinghaus*^{†‡§}, Tobias Hübenthal^{†§}, Jennifer Faber[†]

* Federal Institute for Vocational Education and Training (BIBB), Bonn, Germany,

Email: jens.doerpinghaus@bibb.de, <https://orcid.org/0000-0003-0245-7752>

[†] German Center for Neurodegenerative Diseases (DZNE), Bonn, Germany

[‡] Department of Mathematics and Computer Science, University of Cologne, Germany

[§] These authors contributed equally.

Abstract—This paper presents a novel approach towards link prediction in clinical knowledge graphs. They play a central role in linking data from different data sources and are widely used in big data integration, especially for connecting data from different domains. We present a knowledge graph initially built on data from a clinical trial on Spinocerebellar ataxia type 3 (SCA3), which is a rare autosomal dominant inherited disorder. The contributions of this paper are (1) to create a feasible data representation schema capable of handling clinical imaging data in a knowledge graph and to (2) convert the data efficiently into a knowledge graph. Due to the limited amount of patient-nodes usually common methods for link prediction and graph embeddings are problematic and thus we will (3) present a novel approach for link prediction utilising graph structures and Conditional Random Fields. In addition, we present (4) an extensive evaluation underlining the importance of (a) data management and (b) further research on link prediction using graph structures.

I. INTRODUCTION

KNOWLEDGE graphs have been shown to play an important role in recent knowledge mining settings, for example in the fields of life sciences or bioinformatics. Contextual information is widely used for NLP and knowledge discovery tasks since it highly influences the exact meaning of expressions and also queries on data. Here we will present some results on link prediction in knowledge graphs in the field of personalised medicine which aims for matching certain risk groups and possibly yet unknown subgroups to treatments, ultimately optimising patients' responses, mainly to available drugs. For this purpose, collected primary data of the examined persons have to be linked with data from secondary sources like publications or databases in an application-oriented way.

As part of the European Spinocerebellar Ataxia Type 3 Initiative (ESMI), SCA3 mutation carriers, their first-degree relatives, and healthy controls were prospectively studied using standardised clinical assessment as well as MRI imaging and biosampling.

Spinocerebellar ataxia type 3 (SCA3) is a rare autosomal dominant inherited disorder. The onset of the disease is in adulthood. Patients develop ataxia, which is a disorder of coordination of target movements that affects gait, fine motor skills and speech. The disease is progressive and patients in the advanced stages are usually dependent on the use of first a walking aid and later a wheelchair. Not only the gait

disorder has a strong influence on everyday activities. Also the independent preparation of meals, tool use of e.g. eating utensils and an increasingly unclear speech severely restrict the patients in their everyday life. Although SCA3 mutation carriers are not yet symptomatic, disease activity is already evident, for example, in atrophy of certain areas of the brain where neuropathological changes are predominant, as well as elevated blood levels of non-specific markers for neuron loss. The data set contains not only patient data but also digital imaging data [1], [2].

The goals of this paper are (1) to create a feasible data representation schema capable of handling clinical imaging data in a knowledge graph, see Figure 1, and to (2) convert the data efficiently into a knowledge graph. Since the overall amount of participants in clinical trials is usually not high, employing common methods for link prediction and graph embeddings is problematic [3]. We will (3) present a novel approach to link prediction utilising graph structures and (4) its evaluation.

This paper is divided into six sections. After an introduction, the second section gives a brief overview of the state of the art, related work and backgrounds used for our novel approach. Therefore, we will refer to both knowledge graphs and dedicated algorithms. In the third section, we present our approaches regarding data integration and data schema. The fourth section describes the novel approach to link prediction, with the experimental results on both artificial and real-world scenarios in the subsequent section.

Our conclusions and outlooks are drawn in the final section. We will propose a novel CRF-field based approach which presents promising performance. While the results at first glance do not seem to be a significant improvement for new algorithms for knowledge discovery on clinical data they clearly show the importance of (a) data management and (b) further research on link prediction using graph structures. We also provide a short outlook for extensions of our work.

II. RELATED WORK AND BACKGROUND

Clinical research is more and more relying on data-intensive approaches, thus facing increasingly complex challenges. Expert systems, for example, provide users with several methods for knowledge discovery. They are widely used to find relevant

was generalised to achieve interoperability with clinical data.

In our case, the first step taken is the integration of data from a very complex clinical trial. We will provide a data integration schema in the next section. The data schema should be capable of further data integration, for example *Gene Ontology* or data from scientific documents like *PubMed*. The software importer should be as generic as possible to work on multiple data sources. This helps to provide experimental results on data which is not affected by data protection regulations.

The experimental results are carried out using a Neo4j graph database on a HPC environment utilising parallel learning on several machines. We have provided a generic importer capable of handling different data sources. It makes use of a configurable ini-file which offers a predefined structure and is read-in by the generic importer. All software is available online¹.

III. DATA INTEGRATION AND DATA SCHEMA

The actual data schema for the graph on which this work is based is presented in Figure 2. For this purpose, the data used was first considered, taking into account the underlying data structure. This data structure is formalised and published in the *Registry of DICOM Data Elements*². There the different categories of objects within the *DICOM* metadata are listed, described and linked. The underlying tree structure of the information object definitions (IOD) and their sub-trees consisting of other IODs, modules and attributes is very well displayed in the *DICOM Standard Browser*³. Our data schema was significantly influenced by these sources and represents the inherent data structure using nodes, edges and attributes. The given selection and arrangement of the individual nodes has been made by the author as an exemplary instance. By adjustments in the configuration file also other schemes arise. However, for the graph used in this work it was necessary to decide on a schema. First, it was important to keep the four-level hierarchy of the *DICOM* data. This can be observed in Figure 2 in the middle strand. Each patient has his or her own node linked to his or her studies. These in turn contain the associated series, which then contain the images. In addition to this main strand within the schema, additional information is then annotated. All modules classified as mandatory (M) are included. In addition, at least one module from the classes conditional (C) and user optional (U) was also used. For (C) the class *Contrast/Bolus* is chosen, for (U) we decided on *Patient Study*. Within the data schema, the IOD modules used, which form their own node groups, are highlighted in yellow for visualisation, and the attributes in red. The blue line *Node Group = True* implies that the nodes listed below belong to the node group of the heading. These are represented as triangles in the graph, but are shown here as node groups for clarity. As an example, the *Manufacturer* node

can be considered. It belongs to the node group *General Equipment* and forms a triangle in the graph with the node *General Equipment* and the node *General Series*. An example is shown in Figure 3.3. In contrast, the *General Series* node, for example, has attributes such as *Modality*, *Series Instance UID*, and others stored as node-owned attributes rather than as separate nodes.

However, such specifications can be freely designed and modified via the configuration file, as explained in the section before. In addition to the data contained in the *DICOM* files, two more nodes have been added. *Source* specifies the source of the data. For the given data, this is stored under the tag (0013, 1010). *File* stores the file name of the image and therefore serves as a kind of provenance, allowing the nodes to be uniquely assigned to a respective file. To ensure that said two nodes can be included, it is important for the configuration of the importer that the patient is contained in the graph as a node. This means a minor restriction in the sense of free configuration, however, such a graph without patients should be difficult to justify in terms of content.

Due to data protection rules, we will present results using a second data source, which is open-source and also supports the generic usability of the importer. The *SIMBA Image Management and Analysis System*⁴ is used as our source. From the projects listed there, the *ELCAP Database* and from it again the *Zero-Change Dataset* were selected. The data comes from the *Public Lung Database to Assess Drug Response*, as can be seen from the website. A second configuration file named *dev2.ini* is created for them, which partly contains different nodes from the first one. Since the only purpose is to show conceptually that the script works for other data sets and configurations, only a much smaller total number of node types is used in the configuration file.

IV. LINK PREDICTION

A. Scores based on the topology of the graph

Link prediction belongs to the field of computational analysis of a network, where the nodes represent persons or entities and the edges represent relations. These networks are dynamic and change over time. The link prediction problem deals with a section of such a network at a time t_0 and asks for the most accurate predictions possible for edges that do not yet exist at time t_0 and will be added at a later time t . Among other things, the network's own topology plays a crucial role. To be able to quantify this topology different neighbourhood measures from graph theory and their relative effectiveness are investigated.

In [13] a so-called score is used for the measure of this effectiveness. It is calculated in different ways. Examples are:

- **Common Neighbours:** Given a graph $G = (V, E)$, $Score(x, y) := |\Gamma(x) \cap \Gamma(y)|$ describes the number of common neighbours of two nodes $x, y \in V$. Here, $\Gamma(v)$ denotes the direct neighbourhood of a node $v \in V$. [13]

¹See <https://github.com/TbsHbntHl/master-s-thesis-link-prediction-on-large-scale-knowledge-graphs>.

²See <https://dicom.nema.org/medical/dicom/current/output/chtмл/>

³See <https://dicom.innolitics.com/ciods>

⁴See Simba database - public lung database. <http://www.via.cornell.edu/visionx/simba/>.

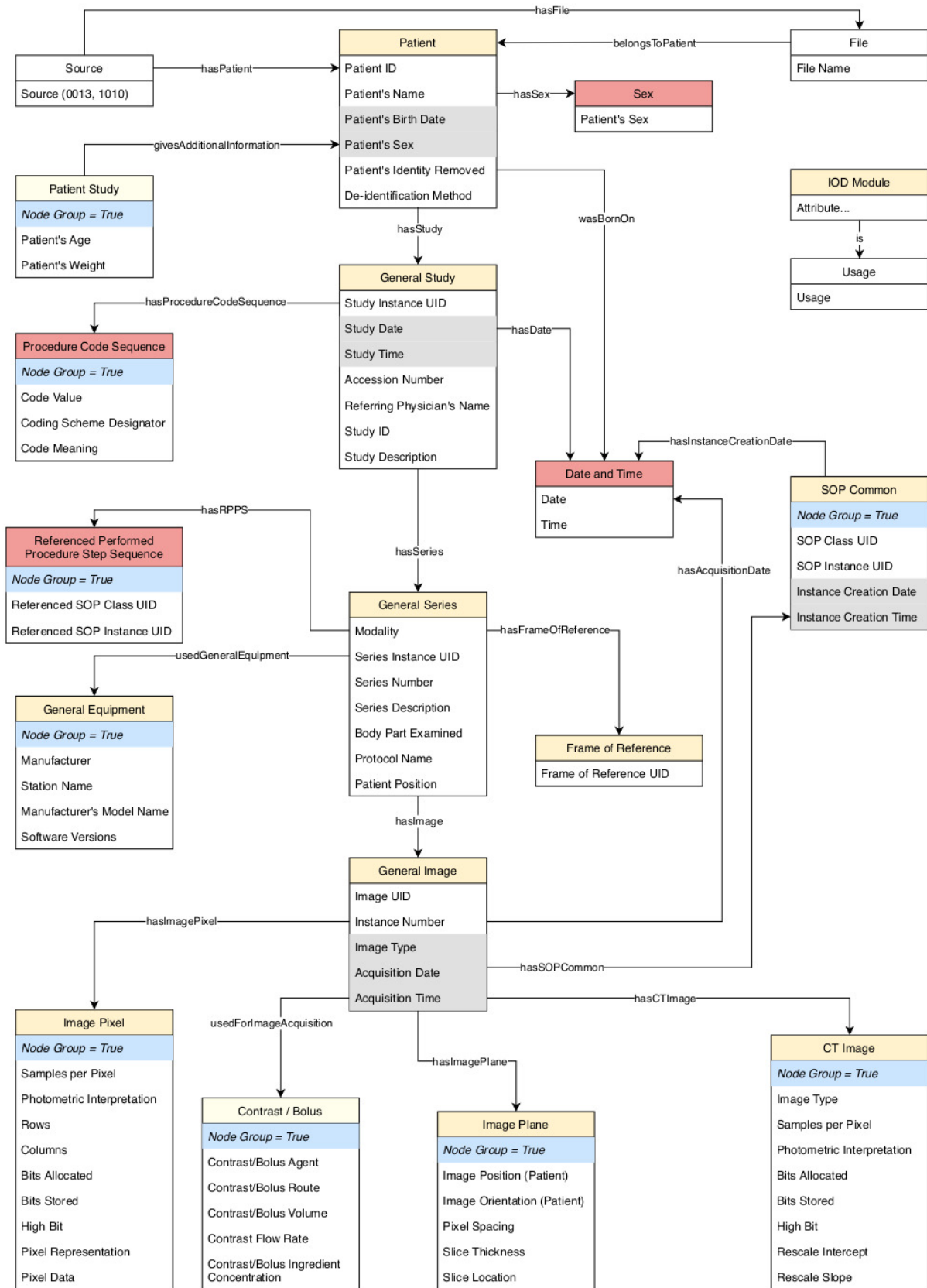


Fig. 2. Data schema for the import of DICOM files.

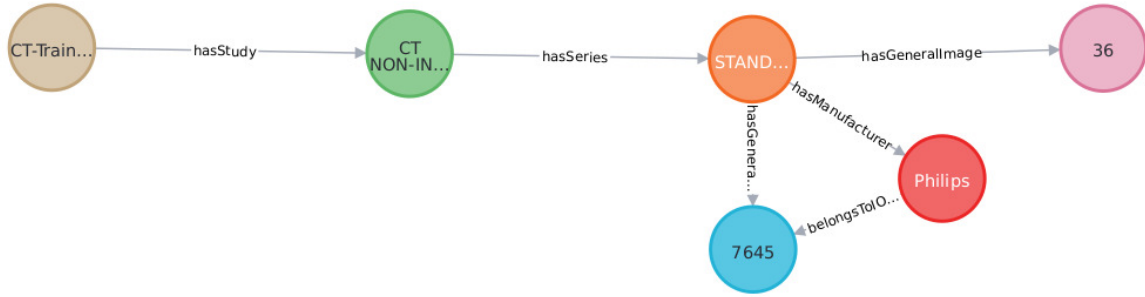


Fig. 3. Partial section of the graph: Exemplary triangle in the graph between the named example nodes Manufacturer (red), General Equipment (blue) and General Series (orange).

- Preferential Attachment: Given again a graph $G = (V, E)$. The underlying premise is the assumption that the probability that a new edge contains the node $x \in V$ is proportional to $|\Gamma(x)|$. Since the measure was originally conceived for predicting future collaborations between two authors, this yields $Score(x, y) := |\Gamma(x)| \cdot |\Gamma(y)|$. This builds on the idea that nodes with many edges have a higher probability of even more edges. [13]
- Adamic/Adar: The coefficient found here originally yields a measure that two homepages are strongly connected. For this purpose, features z are computed from a feature base set F of the two nodes, here web pages, and the commonality is defined as:

$$\sum_{z:\text{features shared by } x,y} \frac{1}{\log(\text{frequency}(z))}$$

This gives less weight to more frequent features than to less frequent ones. If features are to be left out and only the topology of the graph is to be considered, the following score is used for two nodes $x, y \in V$ of a graph $G = (V, E)$:

$$Score(x, y) = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{\log(|\Gamma(z)|)}$$

These measures belong to methods based on node adjacency. [13]

They are presented in the Neo4j database in two ways as the basis of link prediction within the graph used there. First, there is the possibility of making the addition of a new edge conditional on whether the above score exceeds a pre-specified bound. If it does, the edge is added. On the other hand, the scores can be combined with supervised learning: They are used as features to train a binary classifier. This then predicts whether a particular pair of nodes will be connected by an edge with high probability in the future. To train and evaluate the classifier, the graph used is divided into training, testing and validation sets. Then training is performed within the training graph and the result is applied to the test graph. During validation, promising results are shown for the use case. With this work, as will be explained later, a different approach is

taken, but one that also uses these scores as features or as a criterion for choosing a label.

B. Link prediction for paths based on node attributes

The approach adopted in this paper makes use of Conditional Random Fields. Therefore, their origin is briefly examined here and an introduction is given.

a) *Markov chain*: First, a simple Markov chain of order n is considered. The idea is to be able to calculate the probability of future states occurring. The order indicates on how many previous states the next one depends. In a first-order Markov process, the next state depends only on the current state. At the beginning, the system is in the initial state. [19]

Definition IV.1. A Markov process is understood to be a tuple (S, A, δ) . Here S describes the finite set of states, A the set of possible actions, and δ the state transition function. [19]

For each pair (s_t, a_t) with $s_t \in S, a_t \in A$ the state s_t transitions via $\delta(s_t, a_t)$ to the state s_{t+1} . The transitions in this case are usually given in probabilities. The choice of action depends on the current state and can be represented as a function $\pi : S \rightarrow A; \pi(s_t) = a_t$. It is also called a strategy. [19]

b) *Hidden Markov models*: Hidden Markov models are used to represent probability distributions over sequences of observations. A distinction is made between the observation X_t and the state Z_t at time t . The latter is hidden, hence the name of the model. Here, as in the 1-step Markov chains, the so-called Markov property is assumed: Z_t at time t depends only on Z_{t-1} at time $t-1$. An example of this can be seen in Figure 4. The time t need not be an explicit time and can also be implicitly considered as a location within the sequence. The overall probability distribution of a sequence of states and observations can be expressed as an equation as follows:

$$P(Z_{1:N}, X_{1:N}) = P(Z_1)P(X_1|Z_1) \prod_{t=2}^N P(Z_t|Z_{t-1})P(X_t|Z_t)$$

Since the states are hidden and only the observations are considered, which in turn depend on the states, the probability of an N-element sequence is represented by a product of

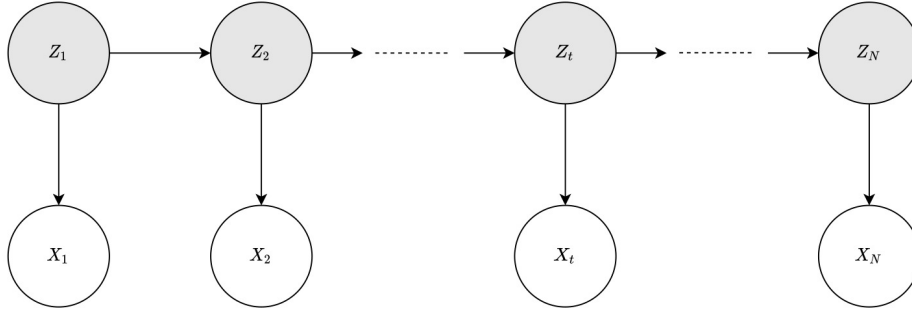


Fig. 4. Example of a hidden Markov model: Z_t describes the state and X_t the observation dependent on it at time t .

conditional probabilities. Moreover, except for the initial state, each state depends on the previous one. [20], [21], [22]

According to [20], [21], there are five elements that characterise a hidden Markov model:

- the number K of states that can be assumed in the model. The states are represented as $K \times 1$ vectors with binary values such that the k -th state at time t takes the value 1 in the k -th row and 0 everywhere else.
- the number Ω of distinct observations that can be observed in the model. Analogous to the states, an $\Omega \times 1$ vector is used.
- the state transition model A : This is also called the state transition probability distribution and describes the probability of changing from a state $Z_{t-1,i}$ to a state $Z_{t,j}$ within one time step. Here $i, j \in 1, \dots, K$. This can be formulated as follows:

$$A_{i,j} = P(Z_{t,j} = 1 | Z_{t-1,i} = 1)$$

Each row of A sums up to 1 in this case.

- the observation model B is an $\Omega \times K$ matrix whose elements $B_{j,k}$ give the probability of making the observation $X_{t,k}$ given the state $Z_{t,j}$:

$$B_{j,k} = P(X_t = k | Z_t = j)$$

- the initial state distribution π is a $K \times 1$ vector with $\pi_i = P(Z_{1,i} = 1)$.

The model is often abbreviated in literature as $\lambda = (A, B, \pi)$. [20], [21]

c) *Markov Random Fields*: Let $G = (V, E)$ be an undirected graph. The nodes $v \in V$ correspond to the random variables which can assume the states. Here, these depend only on the states of the random variables u of their Markov cover $B_v := \{u : (v, u) \in E\}$. This is expressed in the following equation:

$$P(x_1, \dots, x_n) = \frac{1}{Z} \prod_{c \in C} F_c(x_c)$$

Here C is the set of maximal cliques of the graph. The functions F are non-negative and depend on the variables within a clique c . For normalisation, a function $Z = \sum_{x_1, \dots, x_n} \prod_{c \in C} F_c(x_c)$ is used so that the distribution sums up to 1 overall. [23]

d) *Conditional Random Fields*: Conditional Random Fields are a special case of Markov Random Fields and belong to the field of supervised learning. Instead of only considering the probability for a label sequence y , here the probability of a label sequence y , conditioned by an observation sequence x , is determined:

$$P(y|x) = \frac{1}{Z(x)} \prod_{c \in C} F_c(x_c, y_c),$$

$$Z(x) = \sum_{y \in Y} \prod_{c \in C} F_c(x_c, y_c)$$

The normalisation function $Z(x)$ now also depends on x .

In other literature, the definition of a (linear chain) Conditional Random Field is the conditional probability

$$p(y_{1:n} | x_{1:n}) = \frac{1}{Z} \exp \left(\sum_{n=1}^N \sum_{i=1}^F \lambda_i f_i(y_{n-1}, z_n, x_{1:N}, n) \right).$$

Within the exponential function, the first sum is over $n = 1, \dots, N$, which indicates the position of a word, or here a node, within the sequence. The second sum iterates the features f_i weighted by the scalars λ_i , $i = 1, \dots, F$. The values for the weights must be given or learned by the CRF model. They ensure that certain labels are preferred or even avoided. [24] For a given sequence, several features can be active at the same time, i.e., not equal to 0. This is called overlapping features. This can happen because, unlike in hidden Markov models, it is also possible to look at subsequent or previous elements of the sequence. [24] To train, fully labelled training sequences $(x^{(1)}, y^{(1)}), \dots, (x^{(m)}, y^{(m)})$ are required, where $x^{(i)} = x_{1:N}^{(i)} \forall i \in 1, \dots, m$. Thus, the conditional probability of the training data is maximised:

$$\sum_{j=1}^m \log p(y^{(j)} | x^{(j)})$$

This is computed by default employing algorithms that use the gradient descent method. [24]

To assess the quality of the prediction, the F1-score (also balanced F-score or F-measure) is used. This can be regarded

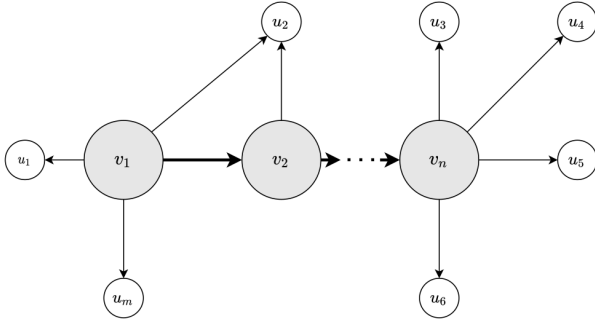


Fig. 5. The input path p consists of the nodes v_i , $i = 1, \dots, n$ highlighted in grey. The white nodes u_j , $j = 1, \dots, m$ serve as labels of the nodes of p .

as a weighted average of the precision and the recall. The best value is 1 and the worst value is 0. The formulas used for this are:

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}},$$

$$\text{Precision} = \frac{TPR}{APR},$$

and:

$$\text{Recall} = \frac{TPR}{APS}.$$

Here TPR means true positive results, APR means all positive results and APS are all samples that should have been identified as positive.

e) Learning Paths: Link prediction is used to predict possible, initially non-existent edges for the previously constructed graph. For this purpose, the graph is imported from *Neo4j* into *Python* via the *py2neo* library⁵. Then, using a query, paths are read in from the graph to be used as input for the conditional random fields and link prediction. The paths are converted to NER-compatible (named entity recognition) form. Thus, a path p is considered first. Then, for each node v contained in p , all neighbouring nodes $u \in \Gamma(v)$ are taken as possible labels. Thus, each node can be used both as a node of a path and as a label for other nodes, see Figure 5.

In the next section, we describe and evaluate different scenarios.

C. Creating one-node paths

The simplest form offers a path of length one, i.e. a single node and its direct neighbourhood. For this purpose, these one-node paths are read from the graph. It is specified which node type is considered, e.g. patients or images. Then a graph query is used to find the direct neighbourhood $\Gamma(v)$ of these nodes $v \in G$ and Γ is stored as a set of labels $l(v)$ for v . Since the CRF library can only assign one label to each node v at a time, criteria must be used for selection. For this purpose, section IV-A is used here to select nodes with, for example,

⁵see <https://py2neo.org/2021.1/>

TABLE I

EXCERPT FROM THE OUTPUT OF QUERY Q1. THE TERM SCORE REFERS TO THE VALUE CALCULATED FOR THE NODE AND ITS LABEL BY NEO4J'S COMMON NEIGHBOURS ALGORITHM.

patientNode	labelNode	score
CT-Training-BE001	SPIE-AAPM Lung CT Challenge	251.0
CT-Training-BE001	1.2.840.113704.1.111.2112.1167842143.1	2.0
CT-Training-BE001g	Patient_Study	2.0
CT-Training-BE001	073Y	1.0
CT-Training-BE001	1-001.dcm	1.0
...		

the highest score in one of the link prediction algorithms available in *Neo4j*. Later, alphabetical sorting is also given as an alternative. The choice of the method for probing the labels on the one hand influences the result and on the other hand also the runtime of the queries. First, single patient nodes are considered. As their label the neighbour with the highest score first at Common Neighbours and then at Total Neighbours is chosen. Afterwards we consider two other nodes, namely General Image and Date. The queries used for this are the following (Since some queries did not terminate they are left out):

```
(Q1) MATCH (p:Patient)-[]-(a) RETURN p.nodeUID
as patientNode, a.nodeUID as labelNode,
gds.alpha.linkprediction.commonNeighbors(p, a)
AS score ORDER BY p.nodeUID, score
DESC, a.nodeUID
```

```
(Q2) MATCH (p:Patient)-[]-(a) RETURN p.nodeUID
as patientNode, a.nodeUID as labelNode,
gds.alpha.linkprediction.totalNeighbors(p, a)
AS score ORDER BY p.nodeUID, score, a.nodeUID
```

```
(Q4) MATCH (p:General_Image)-[]-(a) RETURN
p.nodeUID as imageNode, a.nodeUID as labelNode
ORDER BY p.nodeUID, a.nodeUID
```

```
(Q5) MATCH (p:Date)-[]-(a) RETURN p.nodeUID
as dateNode, a.nodeUID as labelNode ORDER BY
p.nodeUID, a.nodeUID
```

See table I for an example output for query Q1. The algorithm we use for applying the *CRFs* to the paths from the graph consists of the following steps:

Algorithm 1 INTEROPERABLE-DATA

Require: Graph G in *Neo4j*

Ensure: Label prediction, Measurement of prediction success

- 1: readNodePathsFromGraph(G)
 - 2: splitValidationAndTrainingData()
 - 3: for all P in AP :
 - 4: assignFeaturesToNodesInPaths($N(P)$)
 - 5: assignLabelsToNodes($N(P)$)
 - 6: trainUsingCRFs(AP)
 - 7: evaluateResultByComparingToValidationData()
 - 8: **return** predictionVector, F1-Score, Precision, Recall
-

Here $N(P)$ denotes the set of nodes in path P while AP denotes the set of all paths read from the graph. The set of output values consists of a prediction vector as well as the F1 score, the precision and recall.

TABLE II
DETAILS FOR QUERY Q1

node	precision	recall	f1-score	support
SPIE-AAPM Lung CT Chall...	1.0000	1.0000	1.0000	14
accuracy			1.0000	14
macro avg	1.0000	1.0000	1.0000	14
weighted avg	1.0000	1.0000	1.0000	14

V. EVALUATION

A. Runtime

This subsection deals with the consideration of the achieved runtimes of the link prediction programmes. First, the runtime result of the queries Q1 - Q5 is presented. Within the programme, times are measured for all individual sections. The problematic part of the programme is the `sent2features()` method. For illustration the runtime of the time-relevant parts is shown in Figure 6. All other parts of the programme have a negligible very small runtime. This is especially evident for Q4. In this query, the `General Image` node is in the centre, which compared to other nodes such as `Patient` has already got a lot of neighbours due to the structure of the graph. The runtime, which is almost completely generated by `sent2features()`, amounts to a total of slightly less than 11 hours.

B. Quality

The first attempts at link prediction are carried out with single-node paths. Here the focus is initially on the patient. For Q1 link prediction shows the association of the data with the associated study *SPIE-AAPM Lung CT Challenge*. In this case a F1-score of 1 is obtained. This prediction is very accurate, however this is not surprising given the data. The patient node has a very limited type and number of neighbours. Sorting by number of common neighbours leaves only the source. This is also reflected in the detailed look at the labels, as can be seen in Table II.

In these tables, available labels are shown under the heading node. Precision, recall and the F1-score are shown to the right. The value at support indicates the frequency of the find. The opposite results are obtained for sorting by Total Neighbours. Here a F1-score of 0 is obtained. Thus, the prediction has completely failed here. The result can be seen in Table III. Again, the actual result is not surprising considering the data. The patients have different ages and due to the small group of individuals, clustering is unlikely.

The penultimate one-node path query is Q4. Here labels for the node `General Image` are being examined. The label selection is based on alphabetical order. From a biological point of view, a different weighting may be more appropriate, but several methods of label selection should be tried for scientific reasons. For Q4, a nominally very good value of 0.7737 was obtained for the F1-score. The detailed consideration of the result is presented in excerpts in Table IV. It shows that different labels were selected for the images in the prediction, with priority given to the label -1024. For the last query Q5

TABLE III
DETAILS FOR QUERY Q2

node	precision	recall	f1-score	support
1-414.dcm	1.0000	1.0000	1.0000	0.0
1.2.840.113704....	1.0000	1.0000	1.0000	0.0
...				...
060Y	1.0000	0.0000	0.0000	2.0
061Y	1.0000	0.0000	0.0000	1.0
063Y	1.0000	1.0000	1.0000	0.0
...				...
accuracy	0.0000	0.0000	0.0000	8.0
macro avg	0.9730	0.8108	0.7838	8.0
weighted avg	1.0000	0.0000	0.0000	8.0

TABLE IV
DETAILS FOR QUERY Q4

node	precision	recall	f1-score	support
-0.10	1.0000	1.0000	1.0000	0
...				...
-100.70	1.0000	1.0000	1.0000	0
-1000	1.0000	0.0000	0.0000	653
-1000.00	1.0000	1.0000	1.0000	0
...				...
-1024	0.8417	1.0000	0.9141	3786
accuracy	0.8417	0.8464	0.8441	4473
macro avg	0.9988	0.7664	0.7658	4473
weighted avg	0.8660	0.8464	0.7737	4473

there is only a limited set of available nodes and edges of the graph due to the node selection and the given data. The programme nominally returns a very high value with an F1-score of 0.9819. The label predicted for the node `Date` is `SOP_Common`. The values of precision and recall compared to the F1-score are shown for the queries Q1 - Q5 in Figure 7 and Figure 8. The former relates precision and recall to each other. The contour lines provide a visual impression of the corresponding F1-score. The latter shows the three values for precision, recall and F1-score side by side,

VI. CONCLUSION AND OUTLOOK

Our studies pursued several goals. The first and second were to create a feasible data representation schema capable of handling clinical imaging data in a knowledge graph and the generic approach for importing imaging data into a graph. *Neo4j* provides an easy way to import large amounts of data with bulk import and we provide the source code of our solution online. This can be individually configured by the user with the help of the script presented here and the associated configuration file. The design of the graph can be very much defined by the user. For the combination with already existing graphs and data systems an interface can be formed with few lines of code. To do so, only the possibly overlapping node types have to be identified. The corresponding CSV files of the programme presented here can be read in a subsequent programme and the node IDs can be stored in sets. Thus, our solution could also be integrated in analysis workflows, for example utilising text mining.

The third goal was to present a novel approach for link prediction utilising graph structures and applying NER and

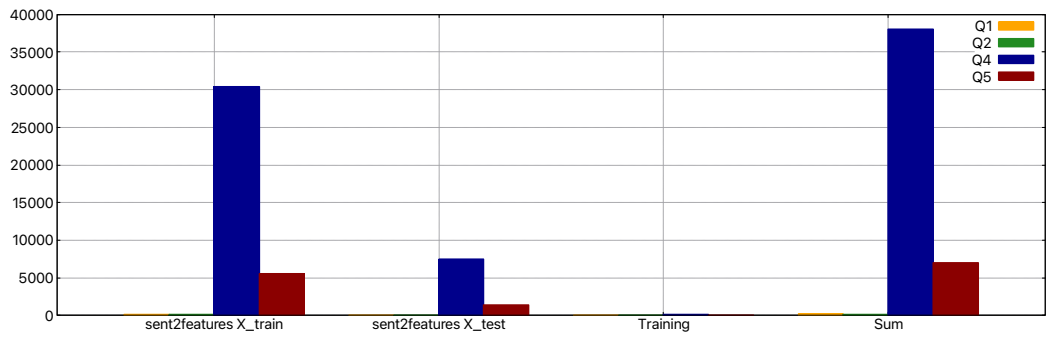


Fig. 6. Average runtime (in seconds) of the relevant parts of the queries Q1 - Q5.

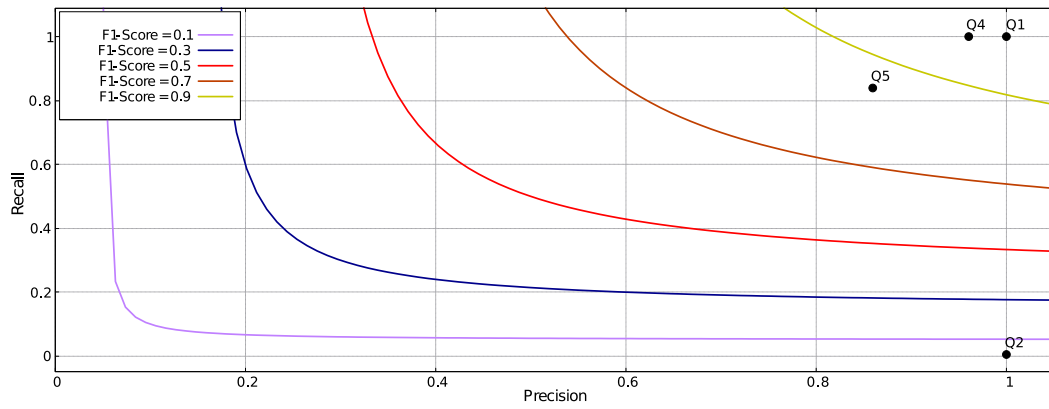


Fig. 7. Precision recall diagram for queries Q1 - Q5.

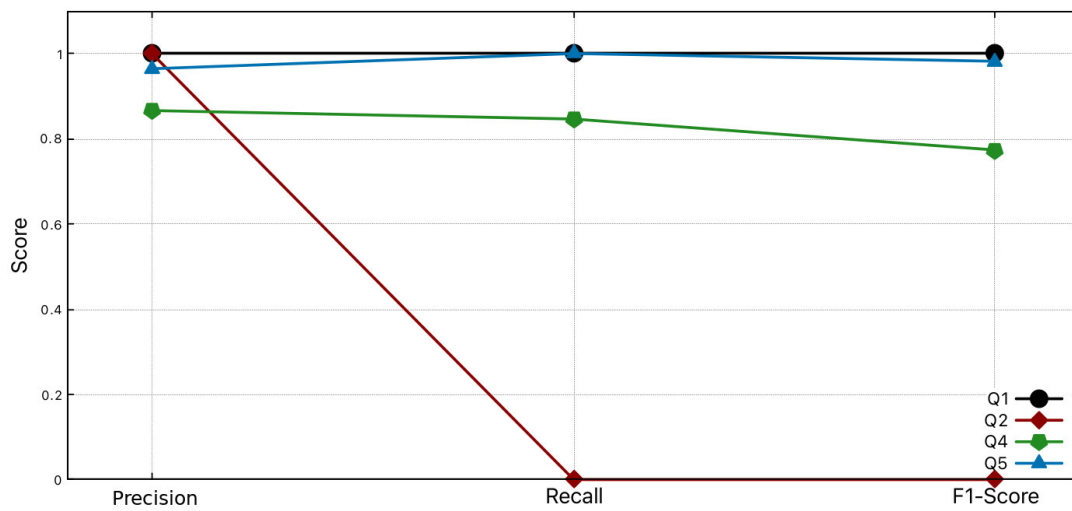


Fig. 8. Comparison of precision, recall and F1-score of queries Q1 - Q5.

CRFs to paths from a graph. For single-node paths, excellent results were obtained for the selected nodes. But we could also show the importance of data management and further research on link prediction using graph structures. For Q1 we could provide trivial results and this clearly underlines the need for data literacy, understanding the structures is essential. Our proposed approach also states the importance of an evaluation with state-of-the-art graph embedding technologies to prove the advantage of keeping graph structures for AI approaches on graphs as [16] proposed.

The next step would be considering multi-node paths which will show an increasing runtime for large data sets. Querying features from the graph in our experimental setting turned out to be very time consuming and scales accordingly with the amount of data. The second problem is the increasing runtime for machine learning as the number of nodes used in the input path grows. At the same time, the requirements for the available main memory also increase enormously. However, both are related not only to the length of the input path, but also to the local environment of the paths. We assume that sparsely populated locations of the graph allow better predictions and provide faster results.

While our proof of concept is both functional and generic, extending the knowledge graph, e.g. with data from text mining on scientific documents, is feasible and just a matter of modelling connectors to the relevant sources since the software is prepared for running in a workflow.

ACKNOWLEDGMENT

We thank the Regional Computing Center of the University of Cologne (RRZK) for providing computing time on the DFG-funded (Funding number: INST 216/512/1FUGG) High Performance Computing (HPC) system CHEOPS as well as support.

REFERENCES

- [1] O. Dössel and T. M. Buzug, *Medizinische Bildgebung*. Walter de Gruyter GmbH & Co KG, 2014.
- [2] D. Peck, "Digital imaging and communications in medicine (dicom): a practical introduction and survival guide," 2009.
- [3] P. Goyal and E. Ferrara, "Graph embedding techniques, applications, and performance: A survey," *Knowledge-Based Systems*, vol. 151, pp. 78–94, 2018.
- [4] C. S. Burns, R. M. Shapiro, T. Nix, J. T. Huber *et al.*, "Examining medline search query reproducibility and resulting variation in search results," *iConference 2019 Proceedings*, 2019.
- [5] A. Callahan, V. Polony, J. D. Posada, J. M. Banda, S. Gombar, and N. H. Shah, "Ace: the advanced cohort engine for searching longitudinal patient records," *Journal of the American Medical Informatics Association*, vol. 28, no. 7, pp. 1468–1479, 2021.
- [6] X. Xu, X. Xu, Y. Sun, X. Liu, X. Li, G. Xie, and F. Wang, "Predictive modeling of clinical events with mutual enhancement between longitudinal patient records and medical knowledge graph," in *2021 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2021, pp. 777–786.
- [7] Hulpus, Ioana and Hayes, Conor and Karnstedt, Marcel and Greene, Derek, "Unsupervised Graph-Based Topic Labelling Using Dbpedia," in *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*, 2013, pp. 465–474.
- [8] J. Dörpinghaus and A. Stefan, "Knowledge extraction and applications utilizing context data in knowledge graphs," in *2019 Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2019, pp. 265–272.
- [9] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig *et al.*, "Gene ontology: tool for the unification of biology," *Nature genetics*, vol. 25, no. 1, p. 25, 2000.
- [10] D. S. Wishart, Y. D. Feunang, A. C. Guo, E. J. Lo, A. Marcu, J. R. Grant, T. Sajed, D. Johnson, C. Li, Z. Sayeeda *et al.*, "Drugbank 5.0: a major update to the drugbank database for 2018," *Nucleic acids research*, vol. 46, no. D1, pp. D1074–D1082, 2017.
- [11] K. Khan, E. Benfenati, and K. Roy, "Consensus qsar modeling of toxicity of pharmaceuticals to different aquatic organisms: Ranking and prioritization of the drugbank database compounds," *Ecotoxicology and environmental safety*, vol. 168, pp. 287–297, 2019.
- [12] H. Cai, V. W. Zheng, and K. C.-C. Chang, "A comprehensive survey of graph embedding: Problems, techniques, and applications," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 9, pp. 1616–1637, 2018.
- [13] D. Liben-Nowell and J. Kleinberg, "The link-prediction problem for social networks," *Journal of the American society for information science and technology*, vol. 58, no. 7, pp. 1019–1031, 2007.
- [14] M. Xu, "Understanding graph embedding methods and their applications," *SIAM Review*, vol. 63, no. 4, pp. 825–853, 2021.
- [15] M. Simonovsky and N. Komodakis, "Graphvae: Towards generation of small graphs using variational autoencoders," in *International conference on artificial neural networks*. Springer, 2018, pp. 412–422.
- [16] W. Cukierski, B. Hamner, and B. Yang, "Graph-based features for supervised link prediction," in *The 2011 International Joint Conference on Neural Networks*. IEEE, 2011, pp. 1237–1244.
- [17] J. Dörpinghaus, A. Stefan, B. Schultz, and M. Jacobs. (2020) Towards context in large scale biomedical knowledge graphs. [Online]. Available: <http://arxiv.org/abs/2001.08392>
- [18] J. Dörpinghaus, V. Weil, S. Schaaf, and T. Hübenenthal, "An efficient approach towards the generation and analysis of interoperable clinical data in a knowledge graph," in *2021 16th Conference on Computer Science and Intelligence Systems (FedCSIS)*. IEEE, 2021, pp. 59–68.
- [19] J. Frochte, *Maschinelles Lernen: Grundlagen und Algorithmen in Python*. Carl Hanser Verlag GmbH Co KG, 2019.
- [20] Z. Ghahramani, "An introduction to hidden markov models and bayesian networks," in *Hidden Markov models: applications in computer vision*. World Scientific, 2001, pp. 9–41.
- [21] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.
- [22] L. Rabiner and B. Juang, "An introduction to hidden markov models," *IEEE ASSP Magazine*, vol. 3, no. 1, pp. 4–16, 1986.
- [23] A. Blake, P. Kohli, and C. Rother, *Markov random fields for vision and image processing*. MIT press, 2011.
- [24] X. Zhu, "Cs838-1 advanced nlp: Conditional random fields," *Technical report, The University of Wisconsin Madison*, 2007.

Deep Learning Transformer Architecture for Named-Entity Recognition on Low-Resourced Languages: State of the art results

Ridewaan Hanslo
University of Pretoria
Gauteng, South Africa
Email: ridewaan.hanslo@up.ac.za

Abstract—This paper reports on the evaluation of Deep Learning (DL) transformer architecture models for Named-Entity Recognition (NER) on ten low-resourced South African (SA) languages. In addition, these DL transformer models were compared to other Neural Network and Machine Learning (ML) NER models. The findings show that transformer models substantially improve performance when applying discrete fine-tuning parameters per language. Furthermore, fine-tuned transformer models outperform other neural network and machine learning models on NER with the low-resourced SA languages. For example, the transformer models obtained the highest F-scores for six of the ten SA languages and the highest average F-score surpassing the Conditional Random Fields ML model. Practical implications include developing high-performance NER capability with less effort and resource costs, potentially improving downstream NLP tasks such as Machine Translation (MT). Therefore, the application of DL transformer architecture models for NLP NER sequence tagging tasks on low-resourced SA languages is viable. Additional research could evaluate the more recent transformer architecture models on other Natural Language Processing tasks and applications, such as Phrase chunking, MT, and Part-of-Speech tagging.

Index Terms—Named-Entity Recognition, Natural Language Processing, Neural Networks, Sequence Tagging, XLM-R, Machine Learning, Transformer Models.

I. INTRODUCTION

NATURAL Language Processing (NLP) which has been in existence for more than 70 years, is a branch of Artificial Intelligence [7]. NLP uses computational techniques for the analysis and representation of naturally occurring texts to achieve human-like language processing for various applications and tasks. Machine Translation (MT) was the first computer-based NLP application [7]. Thereafter, applications utilizing NLP such as Information Retrieval, Information Extraction (IE), and Question-Answering (QA) were introduced [7]. These IE applications include sequence tagging tasks such as Named-Entity Recognition (NER) and Part-of-Speech (POS) tagging.

NER is a task that processes natural language, classifying and grouping, for example, words into categories (also known as phrase types) [20]. With the advent of big data and large datasets, classifying natural language in these datasets has become increasingly important. For example, organiza-

tions are able to apply NER in customer support, content classification, and search and recommendation engines [21]. Furthermore, NER findings may be transferred to other NLP tasks such as MT, automatic text summarization, and knowledge base construction [20]. Lack of data severely impedes performance on NER tasks with low-resourced languages [20].

Recently, within NLP research, the use of Neural Network (NN) architectures, also referred to as Deep Learning (DL) architectures, has generated state-of-the-art results for MT, IE, and QA tasks [2], [8]. NN has seen several additions in the past few decades, from Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) to Transformer architectures [3], [4], [8]. CNN is an extensively studied DL architecture inspired by the visual perception mechanisms of living creatures [11]. RNN is concerned with sequential data that display correlations between data points within a time sequence [12]. Transformers are prominent NN architectures in NLP research, surpassing RNN and CNN in model performance [13].

Transformers facilitate the creation of high-capacity models that are pre-trained on large corpora. These transformers capture long-range sequence features that facilitate parallel training, and the pre-trained models are easily adapted to specific tasks with good performance [13]. XLM-Roberta (XLM-R) is a recent transformer model that has reported state-of-the-art results for NLP tasks and applications, such as NER, POS tagging, phrase chunking, and MT [2], [9].

NLP sequence tagging tasks such as NER and POS tagging have been extensively researched [1]-[7], [9], [10]. However, within the past few years, new DL transformer architecture models such as XLM-R, Multilingual Bidirectional Encoder Representations from Transformers (M-BERT), and Cross-Lingual Language Model (XLM) lower the time needed to train large datasets through greater parallelization. This allows low-resourced languages to be trained and tested with less effort and resource costs while achieving state-of-the-art results for sequence tagging tasks [1], [2], [9]. M-BERT, a single language model pre-trained from monolingual corpora, performs cross-lingual generalization very well [14]. Furthermore, M-BERT is capable of capturing multilingual representations [14]. On the other hand, XLM pre-training has led to solid

This work is based on the research supported by the National Research Foundation of South Africa (Grant Number 138325).

improvements in NLP benchmarks [15]. Additionally, XLM models have contributed to significant improvements in NLP studies involving low-resourced languages [15]. These transformer models are usually trained on very large corpora with datasets that are terabytes (TB) in size.

II. BACKGROUND

A recent study by [1] researched whether NN's are viable for NLP sequence tagging (POS tagging and NER) and sequence-to-sequence (Lemmatization and Compound Analysis) tasks for resource-scarce languages. These resource-scarce languages are ten of the 11 official South African (SA) languages, with English being excluded. The languages are considered low-resourced, with Afrikaans (af) being the more resourced of the ten [1], [10]. This recent study compared two Bidirectional Long Short-Term Memory with Auxiliary Loss (bi-LSTM-aux) NN models to a baseline Conditional Random Fields (CRF) model. The annotated data used for the experiments are derived from the National Centre for Human Language Technology (NCHLT) text project. The results suggest that NN architectures such as bi-LSTM-aux are viable for sequence tagging tasks for most SA languages [1]. However, within the study by [1], NN's did not outperform the CRF Machine Learning (ML) baseline NER model. Rather the CRF model performed better on NER than the bi-LSTM-aux models. Loubser and Puttkammer [1], therefore, advised further studies to be conducted using NN transformer models on resource-scarce SA languages. Additionally, because of the considerable variation in performance per language during their study, [1] suggests conducting further research on the variation in performance per language.

Similarly, a previous study by [18] evaluated XLM-R transformer models for NER on low-resourced languages. However, the fine-tuning of the transformer models was at the model level and not the language level. In other words, a transformer model was fine-tuned on, for example, the Afrikaans (af) language. Thereafter, the model with the fine-tuned parameters for the Afrikaans (af) language was applied to the other remaining nine SA languages. The reason for this decision was due to resource capacity constraints. As a result, the study by [18] produced only a couple higher F-scores than the CRF and bi-LSTM-aux baseline models. Albeit, the CRF model retained the highest average F-score for the ten languages.

For this reason, this study builds upon these previous studies by evaluating the performance of DL transformer architecture for NER on low-resourced languages with fine-tuning of the model applied at the language level. Therefore, the purpose of this study is to evaluate the performance of the NLP NER sequential task using two XLM-R transformer models applying fine-tuning to each model and language combination. In addition, the experiment results are compared to previous research findings.

A. Research Hypotheses

H₁ – There is a performance improvement for NER on the low-resourced SA languages using fine-tuned XLM-R transformer models.

H₂ – Fine-tuned XLM-R transformer models outperform other neural network and machine learning models on NER with the low-resourced SA languages.

B. Paper Layout

The remainder of this paper comprises of the following sections: Sect. III provides information on the languages and datasets while Sect. IV presents the language model architectures. The experiment settings are presented in Sect. V. The results and a discussion of the research findings are provided in Sect. VI and Sect. VII respectively. Section VIII concludes the paper, providing practical implications, limitations of this study and recommendations for future research.

III. LANGUAGES AND DATASETS

As mentioned by [1], SA is a country with at least 35 spoken languages. Of those languages, 11 are granted official status. The 11 languages can further be broken up into three distinct groups. The two West-Germanic languages, English and Afrikaans (af). Five disjunctive languages, Tshivenda (ve), Xitsonga (ts), Sesotho (st), Sepedi (nso) and Setswana (tn) and four conjunctive languages, isiZulu (zu), isiXhosa (xh), isiNdebele (nr) and Siswati (ss). A difference between SA disjunctive and conjunctive languages is the former has more words per sentence than the latter. Therefore, disjunctive languages have a higher token count than conjunctive languages. For further details on conjunctive and disjunctive languages and examples thereof, see [1].

The datasets for the ten evaluated languages are available from the South African Centre for Digital Language Resources online repository (<https://repo.sadilar.org/>). These annotated datasets are part of the NCHLT Text Resource Development Project, developed by the Centre for Text Technology (CTeX, North-West University, South Africa) with contributions by the SA Department of Arts and Culture. The annotated data is tokenized into five phrase types. These five phrase types are:

1. LOC - Location
2. MISC - Miscellaneous
3. ORG - Organization
4. OUT - not considered part of any named-entity
5. PER - Person

The LOC, ORG and PER phrase types are entity names and are the main named entity category used in this study. The MISC phrase type as explained by [10] are for phrase types that form part of either the number expressions or temporal named entity categories so as not to lose the opportunity to annotate the data, which can be further annotated in the future [10].

It is important to note that this annotated data is the same dataset used by the previous studies [1], [10], [18]. However, the studies by [10] and [18] clearly indicates the inclusion of the MISC phrase type whereas, the study by [1] does not.

The previous studies made use of the CoNLL-2003 shared task protocol for data tagging [19]. Additionally, the named entities are further annotated with the beginning [B], inside [I], and outside [O] labelling scheme, which is posited to be ideal for sequence tagging training [10].

The datasets consist of SA government domain corpora. Therefore, the SA government domain corpora are used to do the experiments and comparisons. Eiselen [10] provides further details on the annotated corpora.

IV. LANGUAGE MODEL ARCHITECTURES

A. XLM-R

XLM-Roberta (XLM-R) is a transformer-based multilingual masked language model [2]. This language model trained on 100 languages uses 2.5 TB of CommonCrawl (CC) data [2]. From the 100 languages used by the XLM-R multilingual masked language model, it is noted that Afrikaans (af) and isiXhosa (xh) are included in the pre-training.

As indicated by [2], the benefit of this model is training the XLM-R model on cleaned CC data increases the amount of data for low-resourced languages. Further, because the XLM-R multilingual model is pre-trained on many languages, low-resourced languages improve performance due to positive transfer [2].

The study by [2] reports the state-of-the-art XLM-R model performs better than other NN models such as mBERT and XLM on QA, classification, and sequence labelling. For this research study, two transformer models are used for NER evaluation. The XLM-R_{Base} NN model and the XLM-R_{Large} NN model. The XLM-R_{Base} model has 12 layers, 768 hidden states, 12 attention heads, 250 thousand vocabulary size, and 270 million parameters. The XLM-R_{Large} model has 24 layers, 1024 hidden states, 16 attention heads, 250 thousand vocabulary size, and 550 million parameters [2]. Both pre-trained models are publicly available (<https://bit.ly/xlm-rbase>, <https://bit.ly/xlm-rlarge>).

These pre-trained models (XLM-R_{Base} and XLM-R_{Large}), as mentioned in Sect. I allow low-resourced languages to be trained and tested with fewer resource costs and effort. Therefore, they were fed into this study's DL transformer architecture NER model as part of the NER evaluation process. The model was developed with the Python programming language, the PyTorch ML framework, the Facebook AI Research Sequence-to-Sequence Toolkit (written in Python), and the PyTorch Transformers library. The developed model incorporated the AdamW PyTorch algorithm (optimizer) with warm-up scheduling was trained, validated and then evaluated on the test data. Section V discusses the model's experimental settings.

B. CRF

Conditional Random Fields (CRF) is used for building probabilistic models for segmentation and labelling of sequence data [5]. CRF as ML models are simple, yet, successfully used for NLP sequence tagging tasks, such as NER and POS tagging [4]. Before the use of CRF, Hidden Markov Models (HMM) and stochastic grammars were widely used probabilistic models for tagging tasks [5]. The benefit of using CRF as a sequence modelling framework is it addresses label biases much better than HMM [5]. Additionally, CRF also provides for better stochastic context-free grammar generalization. More information on the CRF ML model is provided by [5].

Eiselen [10] used a CRF ML model for NER on the ten low-resourced SA languages, and [1] included this model as the baseline to compare their two bi-LSTM-aux NN architecture models. Loubser and Puttkammer's [1] findings show that the bi-LSTM-aux models were almost on par with the CRF model, meaning that the NN models did not outperform the ML model. For this reason, this study includes the CRF model as it would be good to compare the DL models with ML models that consistently performs well on NLP sequence tasks. The code for this model is publicly available (<https://taku910.github.io/crffp>).

C. bi-LSTM-aux

Bidirectional Long Short-Term Memory with Auxiliary Loss (bi-LSTM-aux) NN models have been reported as successful with NLP sequence modelling tasks [3]. Modelling tasks include POS tagging, NER, sentiment analysis, and dependency parsing [3]. While both LSTM and bi-LSTM models are classified as RNN, bi-LSTMs implement a backward and forward pass through the sequence before proceeding to the next layer within the network [3]. The inclusion of the auxiliary loss function in the model is to help improve performance gains for rare words used within the corpora [3]. The bi-LSTM-aux model was trained on 22 languages, using polyglot embeddings, and data obtained from the Universal Dependencies project [3]. Additional details on the model are obtainable from [3]. The findings from the study by [3] using their novel bi-LSTM-aux model for POS tagging suggests that the model is as effective as HMM and CRF tagging models. Loubser and Puttkammer [1] used bi-LSTM-aux and bi-LSTM-aux with embeddings model variations for their study. The code for these models is publicly available (<https://github.com/bplank/bilstm-aux>).

V. EXPERIMENTAL SETTINGS

The experimental settings for the XLM-R_{Base} and XLM-R_{Large} models are described next, followed by the evaluation metrics and the corpora descriptive statistics.

TABLE I.
FINE-TUNED PARAMETERS PER LANGUAGE AND MODEL COMBINATION

Language	Model	Learning Rate	Warmup Proportion	Dropout
Afrikaans (af)	Base	6e-5	0.0	0.0
	Large	6e-5	0.0	0.2
isiNdebele (nr)	Base	6e-5	0.0	0.0
	Large	6e-5	0.0	0.0
isiXhosa (xh)	Base	6e-5	0.0	0.2
	Large	6e-5	0.1	0.2
isiZulu (zu)	Base	6e-5	0.1	0.2
	Large	6e-5	0.0	0.2
Sepedi (nso)	Base	7e-5	0.1	0.3
	Large	7e-5	0.1	0.3
Sesotho (st)	Base	6e-5	0.0	0.2
	Large	6e-5	0.1	0.2
Setswana (tn)	Base	6e-5	0.1	0.3
	Large	6e-5	0.0	0.0
Siswati (ss)	Base	6e-5	0.0	0.2
	Large	6e-5	0.0	0.2
Tshivenda (ve)	Base	6e-5	0.0	0.2
	Large	6e-5	0.0	0.2
Xitsonga (ts)	Base	6e-5	0.0	0.2
	Large	6e-5	0.0	0.0

A. XLM-R Settings

The training, validation, and test dataset split was 80%, 10%, and 10%, respectively. Table I provides the fine-tuned parameters at the model and language level while the shared settings across the models and languages are as follows:

- Gradient accumulation steps: 4
- Maximum sequence length: 128
- Training batch size: 32
- Training epochs: 10

B. Evaluation Metrics

Precision, Recall and F-score are evaluation metrics used for text classification tasks, such as NER. These metrics are used to measure the model’s performance during the experiments. The formulas for these metrics leave out the correct classification of true negatives (tn) and false negatives (fn), referred to as negative examples, with greater importance placed on the correct classification of positive examples such as true positives (tp) and false positives (fp) [16]. For example, correctly classified spam emails (tp) are more important than correctly classified non-spam emails (tn). In addition, multi-class classification was used for the research experiments to classify a token into a discrete class from three or more classes. The metric’s macro-averages were used for evaluation and comparison. Macro-averaging (M) treats classes equally, while micro-averaging (μ) favors bigger classes [16]. Each evaluation metric and its formula as described by [16] are listed below.

Precision M : “the number of correctly classified positive examples divided by the number of examples labeled by the system as positive” (1).

$$\frac{\sum_{i=1}^l \frac{tp_i}{tp_i + fp_i}}{l} \quad (1)$$

Recall M : “the number of correctly classified positive examples divided by the number of positive examples in the data” (2).

$$\frac{\sum_{i=1}^l \frac{tp_i}{tp_i + fn_i}}{l} \quad (2)$$

Fscore M : “a combination of the above” (3).

$$\frac{(\beta^2 + 1)Precision_M Recall_M}{\beta^2 Precision_M + Recall_M} \quad (3)$$

C. Corpora Descriptive Statistics

Table II provides descriptive statistics for the language’s training data. As mentioned earlier, disjunctive languages have a higher token count than conjunctive languages. Albeit, the unique phrase type and named entity count for conjunctive languages are, on average, higher than the disjunctive languages.

TABLE II.
THE TEN LANGUAGES TRAINING DATA DESCRIPTIVE STATISTICS

Language	Writing System	Tokens	Phrase Types	Named Entities
af	Mixed	184 005	22 693	21 100
nr	Conjunctive	129 577	38 852	25 030
xh	Conjunctive	96 877	33 951	15 185
zu	Conjunctive	161 497	50 114	25 216
nso	Disjunctive	161 161	17 646	19 163
st	Disjunctive	215 655	18 411	19 211
tn	Disjunctive	185 433	17 670	18 993
ss	Conjunctive	140 783	42 111	21 403
ve	Disjunctive	188 399	15 947	14 119
ts	Disjunctive	214 835	17 904	24 376

VI. RESULTS

Fig. 1 depicts the precision scores for the ten low-resourced SA languages under the five NER models. The Afrikaans (af) language has the highest precision score with 81.74% for the XLM-R_{Large} model, while the Sesotho (st) language has the lowest precision score of 50.31% for the bi-LSTM-aux emb model. Table III displays the precision scores of the two XLM-R transformer models compared to

models used by [1] and [10]. The XLM-R models share five of the ten highest precision scores, with the highest average score belonging to the CRF model with 75.64%. The bold scores in Tables III, IV, V and VI show the highest evaluation metric score for each language and the model with the highest average score.

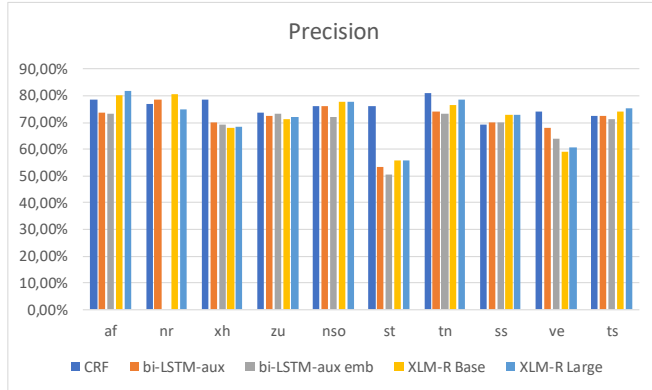


Fig. 1 The precision % for the 10 low-resourced SA languages visual representation

Fig. 2 depicts the recall scores for the ten SA languages under the five NER models. As with the precision evaluation metric, the Afrikaans (af) language has the highest recall scores for three of the five models, with an 87.07% for the XLM-R_{Large} model. Sesotho (st) has the lowest recall score of 55.56% for the bi-LSTM-aux model. Table IV displays the recall scores for the ten low-resourced SA languages. The XLM-R models share the highest recall scores for seven of the ten languages, with the highest average score belonging to the XLM-R_{Base} model with 76.34%.

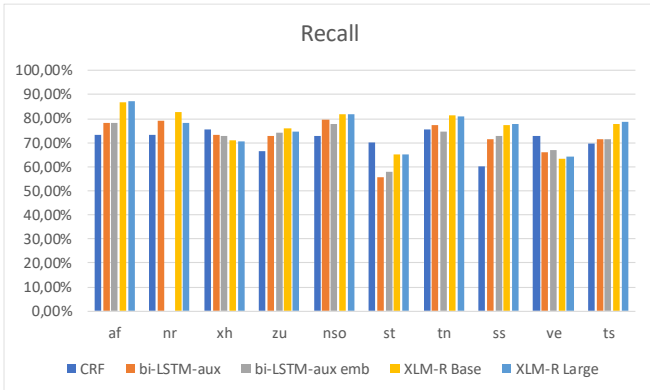


Fig. 2 The recall % for the 10 low-resourced SA languages visual representation

Fig. 3 depicts the F-scores for the ten languages under the five NER models. The Afrikaans (af) language has the highest F-scores for three of the five models, with 84.25% for the XLM-R_{Large} model. Sesotho (st) has the lowest F-score of 53.77% for the bi-LSTM-aux emb model. Table V displays the F-score comparison. The XLM-R models produced the highest F-scores for six of the ten languages, with the highest average score belonging to the XLM-R_{Base} model with 73.64%.

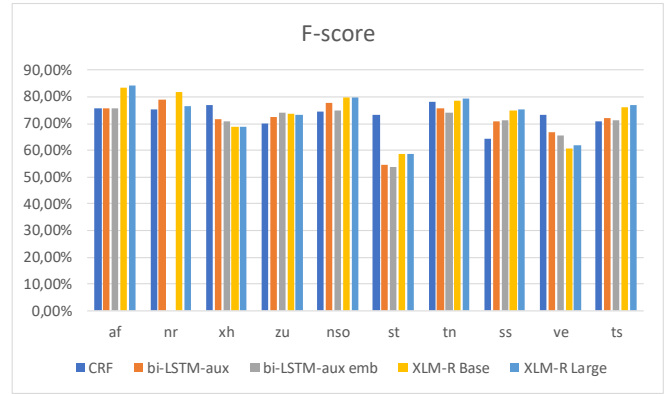


Fig. 3 The F-score % for the 10 low-resourced SA languages visual representation

VII. DISCUSSION

This section discusses the research findings concerning hypotheses testing. The first alternate hypothesis is accepted or rejected based on the XLM-R transformer model’s performance using the three-evaluation metrics. The second hypothesis is accepted or rejected based on the XLM-R transformer model’s performance compared to the CRF and bi-LSTM-aux models used in previous SA NER studies.

H₁ – There is a performance improvement for NER on the low-resourced SA languages using fine-tuned XLM-R transformer models.

The XLM-R_{Large} and XLM-R_{Base} transformer models produced F-scores that ranged from 53.77% for the Sesotho (st) language to 84.25% for the Afrikaans (af) language. In addition, many of the models recall scores were greater than 75% whereas, the precision scores were averaging at 70%. Remember, in this instance, the recall metric emphasizes the average per-named-entity effectiveness of the classifier to identify named entities, and the precision metric compares the alignment of the classifier’s average per-named-entities to the named entities in the data. All F-scores were above 60% except the Sesotho (st) language, which for both XLM-R models were below 60%.

A previous study by [18] was only able to achieve F-scores of 39% for the Sesotho (st) language and proposed that using different hyper-parameter tuning (fine-tuning) and dataset splits could produce higher F-scores. Further, [18] also suggested that further studies could implement the transformer models with discrete fine-tuning parameters per language to produce higher F-scores. The findings of this study show that transformer models with discrete fine-tuning parameters per language generate higher F-scores (see Table VI). The fine-tuned transformer models produced an average F-score 6% higher than the previous transformer models.

TABLE III.
THE PRECISION % COMPARISON BETWEEN TRANSFORMER MODELS AND PREVIOUS SA LANGUAGE NER STUDIES

Precision					
	CRF*	bi-LSTM-aux**	bi-LSTM-aux emb**	XLM-R _{Base}	XLM-R _{Large}
af	78.59%	73.61%	73.41%	80.35%	81.74%
nr	77.03%	78.58%	n/a***	80.74%	74.73%
xh	78.60%	69.83%	69.08%	67.94%	68.46%
zu	73.56%	72.43%	73.44%	71.26%	71.91%
nso	76.12%	75.91%	72.14%	77.90%	77.75%
st	76.17%	53.29%	50.31%	55.67%	55.62%
tn	80.86%	74.14%	73.45%	76.58%	78.65%
ss	69.03%	70.02%	69.93%	72.98%	72.84%
ve	73.96%	67.97%	63.82%	58.85%	60.61%
ts	72.48%	72.33%	71.03%	74.18%	75.15%
Average	75.64%	70.81%	68.51%	71.64%	71.74%

* As reported by [10]. ** As reported by [1]. *** No embeddings were available for isiNdebele.

TABLE IV.
THE RECALL % COMPARISON BETWEEN TRANSFORMER MODELS AND PREVIOUS SA LANGUAGE NER STUDIES

Recall					
	CRF*	bi-LSTM-aux**	bi-LSTM-aux emb**	XLM-R _{Base}	XLM-R _{Large}
af	73.32%	78.23%	78.23%	86.89%	87.07%
nr	73.26%	79.20%	n/a***	82.92%	78.27%
xh	75.61%	73.30%	72.78%	71.05%	70.48%
zu	66.64%	72.64%	74.32%	75.92%	74.58%
nso	72.88%	79.66%	77.63%	81.85%	82.05%
st	70.27%	55.56%	57.73%	65.04%	65.04%
tn	75.47%	77.42%	74.71%	81.38%	80.74%
ss	60.17%	71.44%	72.82%	77.20%	77.88%
ve	72.92%	65.91%	67.09%	63.24%	64.22%
ts	69.46%	71.44%	71.25%	77.99%	78.90%
Average	71.00%	72.48%	71.84%	76.34%	75.92%

* As reported by [10]. ** As reported by [1]. *** No embeddings were available for isiNdebele.

TABLE V.
THE F-SCORE % COMPARISON BETWEEN TRANSFORMER MODELS AND PREVIOUS SA LANGUAGE NER STUDIES

F-score					
	CRF*	bi-LSTM-aux**	bi-LSTM-aux emb**	XLM-R _{Base}	XLM-R _{Large}
af	75.86%	75.85%	75.74%	83.47%	84.25%
nr	75.10%	78.89%	n/a***	81.69%	76.44%
xh	77.08%	71.52%	70.88%	68.85%	68.80%
zu	69.93%	72.54%	73.87%	73.48%	73.17%
nso	74.46%	77.74%	74.79%	79.82%	79.83%
st	73.09%	54.40%	53.77%	58.78%	58.72%
tn	78.06%	75.74%	74.07%	78.70%	79.54%
ss	64.29%	70.72%	71.35%	74.91%	75.19%
ve	73.43%	66.92%	65.41%	60.68%	61.99%
ts	70.93%	71.88%	71.14%	76.03%	76.97%
Average	73.22%	71.62%	70.11%	73.64%	73.49%

* As reported by [10]. ** As reported by [1]. *** No embeddings were available for isiNdebele.

Therefore, the alternative hypothesis is accepted as there is a performance improvement on NER with the low-resourced SA languages using fine-tuned XLM-R transformer models. It is important to note that the previous study by [18] was not included in the comparative analysis with previous SA NER studies because the experimental results were insignificant when compared to the average F-scores of the [1] and [10] studies (see Table V).

H₂ – Fine-tuned XLM-R transformer models outperform other neural network and machine learning models on NER with the low-resourced SA languages.

The fine-tuned transformer models (see Table I) were also compared to the findings of previous studies. In particular, [10] used the CRF ML model to do NER sequence tagging on the ten resource-scarce SA languages. Furthermore, [1] implemented bi-LSTM-aux NN models both with and without embeddings on the same datasets. The comparative

TABLE VI.

THE F-SCORE % COMPARISON BETWEEN TRANSFORMER MODELS AND FINE-TUNED TRANSFORMER MODELS

	F-score			
	XLM-R _{Base} *	XLM-R _{Large} *	XLM-R _{Base}	XLM-R _{Large}
af	82.47%	84.25%	83.47%	84.25%
nr	76.17%	75.60%	81.69%	76.44%
xh	63.58%	64.68%	68.85%	68.80%
zu	72.54%	73.17%	73.48%	73.17%
nso	78.86%	n/a**	79.82%	79.83%
st	38.94%	39.48%	58.78%	58.72%
tn	69.78%	71.91%	78.70%	79.54%
ss	67.57%	68.34%	74.91%	75.19%
ve	60.68%	61.99%	60.68%	61.99%
ts	65.57%	66.12%	76.03%	76.97%
Average	67.61%	67.28%	73.64%	73.49%

* As reported by [18]. ** The model was unable to produce scores for Sepedi.

analysis reveals the performance improvement of implementing DL transformer architecture for NLP sequence tagging tasks such as NER. For example, when analyzing the F-scores, the XLM-R models have the highest F-scores for six of the ten languages, and the CRF model has three of the highest F-scores (see Table V). Meanwhile, the bi-LSTM-aux models had only one of the highest F-scores (see Table V).

This study's result is an improvement for NER research in the SA context because the previous studies by [1] and [18] could not outperform the CRF ML model implemented by [10] until now. Albeit, not all the SA languages are good candidates for DL architectures. For example, the isiXhosa (xh) and Tshivenda (ve) languages consistently underperform compared to the CRF ML model. Additionally, the comparative analysis identified the Sesotho (st) language as the lowest-performing language across the

NN models, with an average F-score of 56%, making it an outlier and an unviable language for current DL architectures for NER.

Therefore, the alternative hypothesis is accepted as fine-tuned XLM-R transformer models outperform other neural network and machine learning models on NER with the low-resourced SA languages (see Table V and Table VI).

This study reveals that the fine-tuned XLM-R transformer models perform relatively well on low-resourced SA languages with NER sequence tagging. Noticeably, there is no distinct performance difference between disjunctive and conjunctive languages. In addition, Afrikaans (af) outperform the other languages using the transformer models. As mentioned earlier, the outlier is the Sesotho (st) language, with the CRF baseline model F-score being 14% more than the XLM-R models and 18% more than the bi-LSTM-aux models. In confirmation with [18], including a language, such as isiXhosa (xh) during the transformer model pre-training does not guarantee good performance during evaluation.

Further, [10] suggested excluding the MISC phrase type to determine whether recall can be improved upon, however, this study revealed that even with the inclusion of the MISC type, the transformer models increased the recall scores considerably. Nonetheless, this is not to say that re-evaluation using an updated list of named entities will not produce higher metric scores.

VIII. CONCLUSION

This research reports on the implementation of Neural Network (NN) and Machine Learning (ML) models to evaluate Named-Entity Recognition (NER) sequence tagging on the ten low-resourced languages of South Africa (SA). The models were trained, validated, and tested using SA government domain corpora. Given the findings, the XLM-R transformer models performed better than the CRF and bi-LSTM models on recall and F-score. The transformer models produced higher F-scores for six of the ten SA languages, while the CRF model had only three of the highest F-scores. The CRF remained dominant on precision, averaging around 75%.

In addition, both the hypotheses were accepted (see Section VII). Firstly, using fine-tuned XLM-R transformer models improves performance on NER for low-resourced SA languages substantially. Secondly, fine-tuned XLM-R transformer models outperform other neural network and machine learning models on NER for the low-resourced SA languages. Therefore, NN transformer models are feasible for sequence tagging tasks, such as NER.

The implications of this study for research and practice of NER using NN transformer models are that such models are not only viable for low-resourced languages but advisable given they require less effort and resource costs. Furthermore, this approach to NER tasks benefits other downstream NLP tasks and applications. These tasks and

applications include question answering, machine translation, and machine reading comprehension.

A limitation of this research is not evaluating the more recent XLM-RXL and XLM-RXXL models on the NER sequence tagging task. Furthermore, the datasets could be re-evaluated using an updated list of named entities.

Additional research could evaluate transformer models on other NLP applications and tasks. Further, NLP tasks and applications could be tested using a linear-complexity recurrent transformer variant and a frozen pre-trained transformer model.

REFERENCES

- [1] Loubser, M., & Puttkammer, M. J. (2020). Viability of neural networks for core technologies for resource-scarce languages. *Information (Switzerland)*. <https://doi.org/10.3390/info11010041>
- [2] Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., & Stoyanov, V. (2020). *Unsupervised Cross-lingual Representation Learning at Scale*. <https://doi.org/10.18653/v1/2020.acl-main.747>
- [3] Plank, B., Søgaard, A., & Goldberg, Y. (2016). Multilingual part-of-speech tagging with bidirectional long short-term memory models and auxiliary loss. *54th Annual Meeting of the Association for Computational Linguistics, ACL 2016 - Short Papers*. <https://doi.org/10.18653/v1/p16-2067>
- [4] Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., & Dyer, C. (2016). Neural architectures for named entity recognition. *2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT 2016 - Proceedings of the Conference*. <https://doi.org/10.18653/v1/n16-1030>
- [5] Lafferty, J., McCallum, A., & Pereira, C. N. F. (2001). Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data.
- [6] Kudo, T. CRF++: Yet another CRF toolkit [Electronic resource]. *GitHub*. <https://github.com/taku910/crfpp>.
- [7] Liddy, E. D. (2001). Natural Language Processing. In *Encyclopedia of Library and Information Science*.
- [8] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*.
- [9] Hedderich, M. A., Adelani, D., Zhu, D., Alabi, J., Markus, U., & Klakow, D. (2020). *Transfer Learning and Distant Supervision for Multilingual Transformer Models: A Study on African Languages*. <https://doi.org/10.18653/v1/2020.emnlp-main.204>
- [10] Eiselen, R. (2016). Government domain named entity recognition for South African languages. *Proceedings of the 10th International Conference on Language Resources and Evaluation, LREC 2016*.
- [11] Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., & Chen, T. (2018). Recent advances in convolutional neural networks. *Pattern Recognition*. <https://doi.org/10.1016/j.patcog.2017.10.013>
- [12] Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*. <https://doi.org/10.1109/78.650093>
- [13] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., & Brew, J. (2019). Transformers: State-of-the-art natural language processing. In *arXiv*. <https://doi.org/10.18653/v1/2020.emnlp-demos>.
- [14] Pires, T., Schlinger, E., & Garrette, D. (2020). How multilingual is multilingual BERT? *ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference*. <https://doi.org/10.18653/v1/p19-1493>
- [15] Conneau, A., & Lample, G. (2019). Cross-lingual language model pretraining. *Advances in Neural Information Processing Systems*.
- [16] Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing and Management, 45*(4). <https://doi.org/10.1016/j.ipm.2009.03.002>
- [17] Kadari, R., Zhang, Y., Zhang, W., & Liu, T. (2018). CCG supertagging via Bidirectional LSTM-CRF neural architecture. *Neurocomputing, 283*. <https://doi.org/10.1016/j.neucom.2017.12.050>
- [18] Hanslo, R. (2021). Evaluation of Neural Network Transformer Models for Named-Entity Recognition on Low-Resourced Languages. *16th Conference on Computer Science and Intelligence Systems, FedCSIS*. <https://doi.org/10.15439/2021F7>
- [19] Sang, E. T. K., & De Meulder, F. (2003). Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL*.
- [20] Chen, S., Pei, Y., Ke, Z., & Silamu, W. (2021). Low-resource named entity recognition via the pre-training model. *Symmetry, 13*(5), 786.
- [21] Gao, S., Kotevska, O., Sorokine, A., & Christian, J. B. (2021). A pre-training and self-training approach for biomedical named entity recognition. *PLoS one, 16*(2), e0246310.

Demand forecasting in the fashion business — an example of customized nearest neighbour and linear mixed model approaches

Joanna Henzel*, Łukasz Wawrowski^{||}, Anna Kubina^{||}, Marek Sikora*, Łukasz Wróbel*

* Silesian University of Technology
Department of Computer Networks and Systems
ul. Akademicka 2A, 44-100 Gliwice, Poland
joanna.henzel@polsl.pl

^{||} Łukasiewicz Research Network – Institute of Innovative Technologies EMAG
ul. Leopolda 31, 40-189 Katowice, Poland
lukasz.wawrowski@emag.lukasiewicz.gov.pl

Abstract—The fashion industry is characterised by the need to make demand forecasts in advance and for highly volatile products for which we often have no sales history at the time the forecasts are made. For this reason, it is necessary to propose forecast mechanisms that can cope with the given conditions. Such forecasts can be based on expert predictions for generalized product categories. In this case, the task of machine learning forecasting methods would be to divide the aggregate prediction into forecasts for individual products, in each colour and size. In the paper, we present several approaches to this specific task. We present the use of the naive method, custom nearest neighbour approach, parametric linear mixed model and an ensemble approach. Overall, the best results we obtained for the ensemble method. Our research was based on real data from fashion retail.

I. INTRODUCTION

DEMAND forecasting in the fashion industry is characterised by specific conditions. In this industry, it is necessary to be able to forecast for a long horizon in time. This is because many products are ordered from other parts of the world, where they are produced on a large scale. Products must be ordered in advance so that the entire process of production, delivery, promotion and distribution to shops in different countries can take place on time.

On the other hand, fashion products are highly variable over time. Rarely are products sold for several seasons. Most of them appear on sale only in one sales season, which translates into a very short sales history for a particular product. This relates to natural seasonality due to changing seasons, but also to trends that can vary significantly from one year to the next.

The combination of the need to order products in advance and the volatility of the products being sold means, that we have to make predictions of demand for products that mostly have not previously been on sale. These are difficult conditions for making forecasts. Sales forecasts could be made by experts based on their domain knowledge, but such experts would

also have difficulty determining future sales of a particular product, in a particular size and colour, each week of the following season. Additionally, with many shops, this would be a very time-consuming process. In this case, a good idea is to use automated forecasting methods, which would be based on statistical models or machine learning models.

The research described in this article dealt with the problem of sales forecasting in the fashion industry. What is important, in our research we obtained sales forecasts for product categories and our task was to build on these general predictions the forecasts for individual products (described by specific product type, colour and size). Sales forecasts were made on weekly aggregates. Forecasts for individual categories were provided to us by a business partner and are proprietary. They are not the subject of this article. However, it should be noted that the forecast data could be provided by an expert. Making assumptions about aggregate sales within more general product categories should be a more manageable and less time-consuming task for an expert.

In this paper, we would like to present several approaches to making demand forecast for the specific products from fashion retail based on higher level forecasts. We present naive method, custom nearest neighbour approach, parametric linear mixed model and an ensemble approach.

II. RELATED WORK

Demand forecasts are the basis of most decisions in supply chain management. Forecasting methods applied to this problem are based on both domain knowledge and historical data analysis. In the former approach, the retailers knowledge is utilised to develop demand prognosis. In the later approach statistical and machine learning based methods are intensively used [1]–[3].

Our research is focused on the data describing the demand in a fashion sector. In this area, research on machine

learning methods application were carried out. In the article [4] the authors present the use of deep neural networks for sales forecasting in fashion industry, especially forecasting the sales of new fashion products. They compare the deep learning approach with other algorithms e.g. Decision Trees, Random Forest, Support Vector Regression, Artificial Neural Networks and Linear Regression. The authors found deep learning models to have good performance, however for some metrics the models were not significantly better than some simpler techniques. It shows that in this sector the simpler and interpretable methods may obtain good results.

Pre-season forecasting in fashion retail was also discussed in [5]. The authors point out, that the typical time-series methods can be introduced for this problem, however in most cases the retailers need to forecast new products, so in historical data there is no time-series linked precisely to the forecast product. It means that the retailers need to base on their intuition and the historical data of similar products. The authors also highlight the need for creating explainable forecasting models. Because of explainability and interpretability, many stakeholders still use a very simple and naive approach for forecasting new products — averaging sales of similar products for the new product. If AI method could be used in this sector it is important to give explanation, how the decision was made.

The authors of [6] focus on fast fashion sales forecasting where the data is limited and within limited time. The authors propose a novel algorithm — Fast Fashion Forecasting (3F) — which combines extreme learning machine (ELM) and the grey model (GM). Extreme learning machine was also used for fashion retailing forecasting in the [7]. In the aspect of Fast-Moving Consumer Goods (FMCG), the authors of [8] showed benefits of applying Machine Learning methods in creating demand forecasting models.

III. PROBLEM STATEMENT

In our research, we were working with the specific problem of forecasting demand for fashion products. From our business partner, we obtained their forecasts of sale for pre-defined categories of products. Our task was to divide these forecasts to the forecasts for each separate product that belongs to the category. A unique product is described by a unique combination of attributes: product type, colour and size. The most important requirement was to use provided forecasts, not to create from scratch forecasts for each separate product. The second important assumption was the horizon of the forecasts. We were working with long-term predictions — forecasts had to be made for 29 weeks ahead. Predictions were made for weekly sale aggregates.

In the obtained dataset, we had real historical sale data from a fashion brand with a shop chain. The dataset contained sale data from January 2016 to November 2021. For the period from April 2021 to November 2021, we got from our business partner not only predictions for the whole category, but also their predictions for unique products. Because of this, we considered this part of a dataset as our test dataset, on which the experiment's result will be calculated. The data before

were our train dataset. In the data, we could observed strong seasonality — big peaks of sales before summer. It should also be noted that data contains sales from COVID-19 pandemic and lock-downs connected with this phenomena. Figure 1 presents demand for an example product.

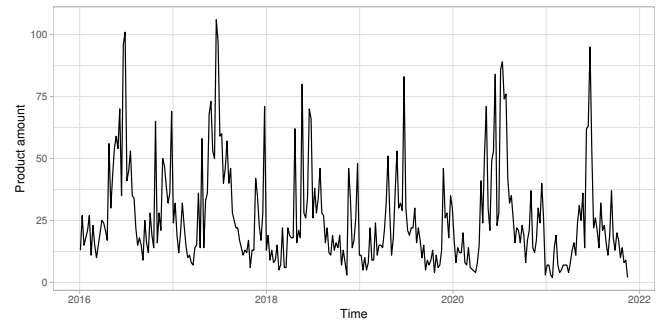


Fig. 1. Demand of selected product

In the data preparation process we created new attributes, that could be defined for new products, that would be in sale many weeks ahead. Because of forecasting demand with a long time-horizon, we couldn't use data about weather or information about sale from weeks just before the forecast week. In our dataset, each observation described one week of a sale of a specific product (a unique product described by a unique combination of attributes: product type, colour and size). Each week, for each product, was described by the attributes listed below.

- Attributes connected with a product in sale: product type, size, main colour and colour undertone.
- Attributes connected with time of a sale (describing the week of an observation): week number, season, quarter of a year, number of days to the closest event (Christmas, Easter, national holiday, Valentine's Day etc.), number of days that passed from the closest event, number of events/holidays that happened during the considered week, number of events/holidays that happened during three weeks — in the week under consideration, the preceding week and following week, binary feature that described if it is the last week of a year.
- Attributes connected with sale: number of sold units, number of sold units within category, fraction of the sale in a given category that account for the sale of the product concerned.
- Attributes connected with trend and seasonality: we used prophet tool [9] to decompose data from train dataset into trend and seasonality for each product. Obtained results were utilized for test dataset.
- ID attributes: product name, category name, date of a first day from a considered week.

In our dataset, we didn't have information about price or planned promotions.

IV. METHODS

In this chapter, we want to present some selected approaches for breaking down the higher level forecasts (forecasts for one category of products) into lower level forecasts (per product forecasts). We present naive method, non-parametric custom nearest neighbour approach, parametric linear mixed model and an ensemble approach. The experiments were performed using R and Python languages.

A. Naive

Firstly, we proposed a naive method of splitting the forecast for the whole category into a forecast for the individual product. This involved finding the weight by which the category forecast was to be multiplied to produce the product forecast.

Suppose the forecast was made for the week n of year Y for a selected product p . The product belongs to category c . In such case, we looked up the sale of the product in week n in the previous year ($Y - 1$). Then, we divided it by the total sales in the category c in week n in the previous year. The given fraction was the weight by which the forecast for the category for week n of year Y was multiplied. The result is the forecast for the product.

An extension of this proposed solution was to determine weight based not only on the week the year before, but also on its week preceding and the week following. This ensured that the weights were averaged and minimized the impact of outliers on the resulting forecast. The forecast for the product was calculated as follows:

$$y_{p,n,Y} = \frac{1}{3} \sum_{i=-1}^1 \frac{s_{p,n+i,Y-1}}{s_{c,n+i,Y-1}} y_{c,n,Y} \quad (1)$$

where s means real sale, y is a forecast, p is a product, for which we calculate the forecast, c is a category of a product, n is a week number, Y is a year. For example, if the year before the forecast week, sale of a product was 2% of total category sale, while the week before it was 6%, and the following week it was 7%, the final weight by which the forecast for the entire category was multiplied was the average of these values — 0.05. A naive method with averaged weights was considered in further stages of the work.

The presented naive approach has one big disadvantage — it works only on a product with history of sale. The naive method could be used for new products only with expert's help. The expert could indicate which product, with historical sales, the new product is similar to. Then, we would assume, that we could use the weight obtained from historical sale of a similar product, to get the forecast for a new product. However, we felt that it would be useful to focus on a solution that would allow us to forecast the demand for new products, with minimal expert involvement in the process.

B. Nearest neighbour (KNN)

As a more advanced forecasting procedure, we propose a nearest neighbour (KNN) approach. This is a non-parametric technique widely used in classification tasks, however we used

it as a method for finding similar observation from historical data. This algorithm, given an input vector, calculates distance (based on a chosen distance metric) to observations from a training dataset. The one observation, whose conditional attribute vector is most similar to the new feature vector, is considered to be its nearest neighbour.

In this approach, we used the dataset described in section III. In order to use some attributes in the KNN approach, additional input data had to be provided. This situation has occurred for "size" attribute. Our dataset included different fashion products, e.g. shirts, bras and socks. These products have different clothing sizes. In order to calculate distance between different observations using the attribute "size", we changed original sizes to numerical values based on our training set, i.e. historical data. We divided the historical dataset by size — a separate subset for each category of products, which have different clothing sizes. Then, for each of the sizes in the set, we determined the percentile values within those subsets. That is, if we forecasted sales for socks in size 36/38, we converted size 36/38 into the numerical value — percentile that this size represented in the historical data.

Additional input data was also provided for attribute "product type". This attribute had nominal values, that were identifiers of the fashion types — the values did not hold any additional meaning. In order to calculate distance between different product types, we proposed using *product types dissimilarity table*. In our *product types dissimilarity table*, we provided the distance between different product types. The distance was equal to 0, if we calculated the distance between products that had the same product type. The distance was equal to 0.5, if we calculated the distance between products that belonged to the same clothe category but are of different product type, e.g. both products were shirts, but they were of different type. The distance was equal to 1, if we calculated the distance between products that belonged to different clothe category, e.g. one product is a shirt and the other is socks.

In the KNN approach, we used the following attributes: week, product type, size, main colour, colour undertone, season, quarter of a year, number of days to the closest event, number of days that passed from the closest event, number of events/holidays that happened during the considered week, number of events/holidays that happened during the week under consideration, the preceding week and following week.

The schema of the proposed procedure is presented in Figure 2. The KNN method is called with the parameter $k=3$. This parameter value was selected from the set of values $k=\{1,2,3\}$, as the value giving the best model results.

It can be noted that this method is based on a similar assumption as our naive method. In the naive approach, we looked for the weight in the history of a given product exactly one year before. In the KNN method, the weight is determined in the same way — it is a fraction of the product's sales to sales in the entire category. The difference is that in the KNN method, we refer to a week that did not necessarily occur exactly one year earlier. Additionally, the nearest neighbour (the nearest "week") may refer to historical data for a different

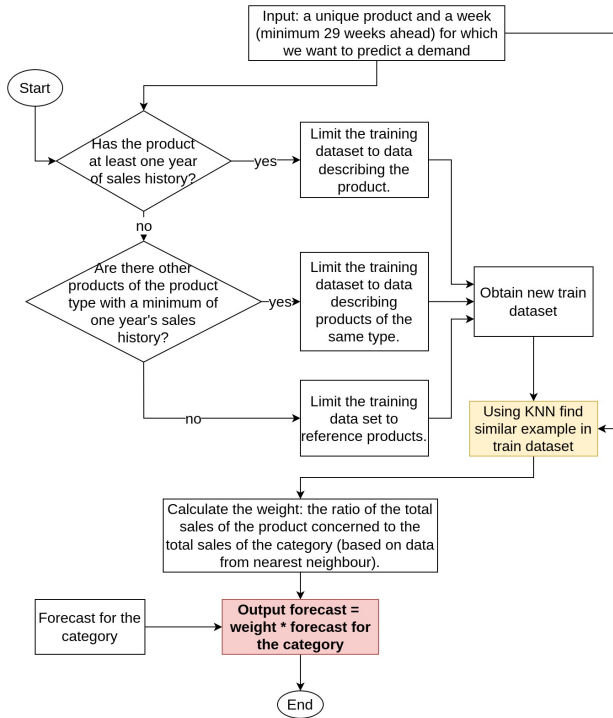


Fig. 2. The schema of the custom nearest neighbour procedure for obtaining the forecasted demand of a product.

product than the one for which the forecast is made.

C. Linear mixed model (LMM)

Linear mixed models are an extension of simple linear regression models and can be used for data with a hierarchical structure which is observed in fashion. These models incorporate fixed and random effects:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\mathbf{u} + \boldsymbol{\varepsilon} \quad (2)$$

where \mathbf{y} is a vector of outcome variable, \mathbf{X} is a matrix of predictors, $\boldsymbol{\beta}$ is a vector of fixed-effects regression coefficient, \mathbf{Z} is a design matrix for random effects, \mathbf{u} is a vector of random effects and $\boldsymbol{\varepsilon}$ is a vector of residuals [10].

In this approach, a model for each product group was estimated with random effects defined by product, size, and color. In groups where there was only one product, random effects were estimated only for size and color. Due to strong asymmetry of outcome variable (forecast weight) Box-Cox transformation [11] was applied.

We assure that there are significant differences between outcome variable and groups defined by product, size, and color using ANOVA. The significance of random effects was confirmed based on the permutation test.

As predictors in linear mixed models, we used attributes connected with product in the sale, time of a sale, trend, and seasonality (described in section III).

D. Ensemble approach

In the ensemble approach, we combined the results from the non-parametric KNN method with the results from the parametric linear mixed model. The forecast returned was the average of forecasts provided by these two models.

V. RESULTS

In this section, we present results obtained using methods described in section IV. We predicted demand for 7 categories of products and aimed for a reduction of mean absolute error (MAE) relative to baseline. Detailed results are presented in table I.

TABLE I
MEAN ABSOLUTE ERROR OF PREDICTED DEMAND IN TEST DATASET

Product category	Baseline	Naive	KNN	LMM	Ensemble
Category 1	3.23	3.77	3.67	3.52	3.59
Category 2	9.68	11.61	11.01	9.29	9.72
Category 3	6.86	7.04	7.12	6.96	6.71
Category 4	6.44	8.43	8.64	7.74	8.12
Category 5	20.23	17.48	20.33	21.09	19.77
Category 6	11.55	11.51	11.48	11.05	10.70
Category 7	4.17	4.70	4.46	4.04	4.07

For the second and seventh categories the best results were obtained for LMM method. For categories third and sixth the lowest errors were obtained for ensemble approach. In the sixth category, we obtain gain in precision for all methods. In the case of the seventh category, both LMM and ensemble approaches have lower errors. For the fifth category also naive approach performed better than the baseline. The first and fourth categories were a little bit problematic because any of the proposed methods do not perform better than baseline. Overall, the biggest improvement is observed for the ensemble methods because for 4 from 7 categories this approach gave better results than the baseline.

In the next step, we tried to investigate what could be a possible reason for such results. Figure 3 presents differences between real and predicted demand for each method and category.

We observed that the fourth and fifth categories were characterized by a much lower number of observations compared to the rest categories. The small sample size could be one of the reasons why it was impossible to obtain a lower MAE than the baseline result for the fourth category. In the case of the fifth and sixth categories, we noticed a few outlier values and a flattened distribution of these errors compared to other categories. In the rest categories, the average of differences between true and predicted demand is close to 0.

The last part of the study was the analysis of stability over time. The Figure 4 shows differences between real demand (bold black line) and forecasts obtained with different approaches for one selected product.

In this particular case, the lowest MAE was obtained for the LMM approach. We can observe that almost all methods predicted lower than actual demand just before summer. The best estimation in this period was observed for KNN method,

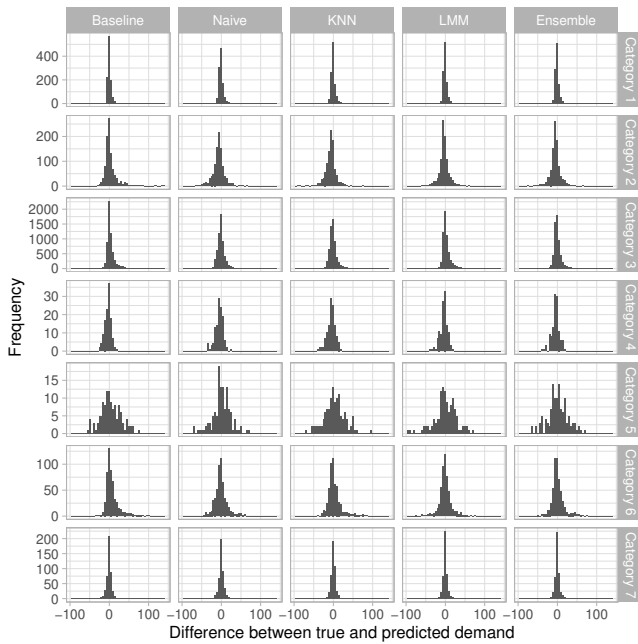


Fig. 3. Distribution of differences between true and predicted demand

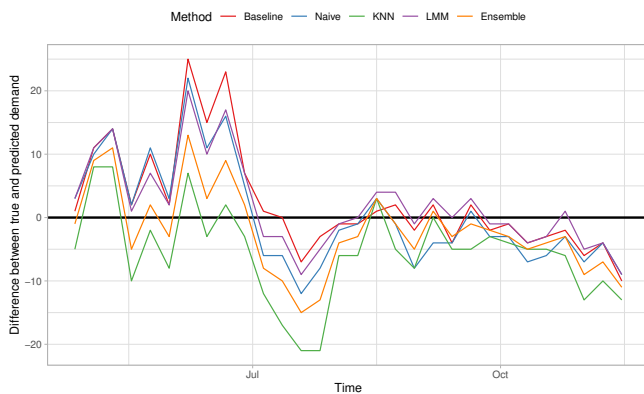


Fig. 4. Differences between true and predicted demand for selected product

however it gives the biggest error in summer. Generally, the proposed approaches underestimate seasonal peaks for all products in our dataset.

VI. CONCLUSIONS AND FUTURE WORKS

Demand forecasting in the fashion business could be problematic due to short product history — in many cases, product is sold for one season. In our case we have access to longer time series nevertheless this data contains unexpected issues connected with COVID-19 such as lock-downs and changing habits of customers.

The proposed solution was based on the share of sale estimation and multiplication of the result by the provided forecast for the category. In almost all cases, utilized methods gave better results than the baseline provided by our business

partner. However, it should be noted that in such an approach we can highlight two sources of possible error — firstly at the method level connected with historical data and predictions and secondly at the category forecast level which differ from real category demand.

Future works explore further the topic of demand forecasting in various business cases: what to do in case of short historical data or how to forecast demand for a totally new products using existing data. We will also investigate other statistical methods designed for this purpose and optimize scope of conditional attributes used by techniques presented in this work.

ACKNOWLEDGMENT

The work was carried out in part within the project co-financed by European Funds entitled “Decision Support and Knowledge Management System for the Retail Trade Industry (SensAI)” (POIR.01.01.01-00-0871/17-00). The research leading to these results received funding by Young Researchers funds of Department of Computer Networks and Systems, Faculty of Automatic Control, Electronics and Computer Science, Silesian University of Technology, Gliwice, Poland (project no.: 02/120/BKM22/0021). This work was partially supported by the European Union through the European Social Fund (grant POWR.03.05.00-00-Z305).

REFERENCES

- [1] M. Z. Babai, J. E. Boylan, and B. Rostami-Tabar, “Demand forecasting in supply chains: a review of aggregation and hierarchical approaches,” *International Journal of Production Research*, vol. 60, no. 1, pp. 324–348, 2022.
- [2] E. Hofmann and E. Rutschmann, “Big data analytics and demand forecasting in supply chains: a conceptual analysis,” *The International Journal of Logistics Management*, 2018.
- [3] M. Seyedan and F. Mafakheri, “Predictive big data analytics for supply chain demand forecasting: methods, applications, and research opportunities,” *Journal of Big Data*, vol. 7, no. 1, pp. 1–22, 2020.
- [4] A. L. Loureiro, V. L. Miguéis, and L. F. da Silva, “Exploring the use of deep neural networks for sales forecasting in fashion retail,” *Decision Support Systems*, vol. 114, pp. 81–93, oct 2018. doi: 10.1016/j.dss.2018.08.010
- [5] S. Sajja, N. Aggarwal, S. Mukherjee, K. Manglik, S. Dwivedi, and V. Raykar, “Explainable AI based Interventions for Pre-season Decision Making in Fashion Retail,” in *ACM International Conference Proceeding Series*, 2020. doi: 10.1145/3430984.3430995. ISBN 9781450388177 pp. 281–289.
- [6] T. M. Choi, C. L. Hui, N. Liu, S. F. Ng, and Y. Yu, “Fast fashion sales forecasting with limited data and time,” *Decision Support Systems*, vol. 59, no. 1, pp. 84–92, mar 2014. doi: 10.1016/j.dss.2013.10.008
- [7] M. Xia, Y. Zhang, L. Weng, and X. Ye, “Fashion retailing forecasting based on extreme learning machine with adaptive metrics of inputs,” *Knowledge-Based Systems*, vol. 36, pp. 253–259, dec 2012. doi: 10.1016/j.knsys.2012.07.002
- [8] “Machine learning in predicting demand for fast-moving consumer goods: An exploratory research,” *IFAC-PapersOnLine*, vol. 52, no. 13, pp. 737–742, 2019. doi: 10.1016/j.ifacol.2019.11.203
- [9] S. J. Taylor and B. Letham, “Forecasting at scale,” *The American Statistician*, vol. 72, no. 1, pp. 37–45, 2018.
- [10] J. Fox, *Applied regression analysis and generalized linear models*. Sage Publications, 2015.
- [11] G. E. Box and D. R. Cox, “An analysis of transformations,” *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 26, no. 2, pp. 211–243, 1964.

Improving Re-rankCCP with Rules Quality Measures

Piotr Jezusek

Faculty of Electronics, Telecommunications and Informatics,
 Gdańsk University of Technology, Gdańsk, Poland

Aleksandra Karpus

Faculty of Electronics, Telecommunications and Informatics,
 Gdańsk University of Technology, Gdańsk, Poland
 Email: alekarpu@pg.edu.pl

Abstract—Recommender Systems are software tools and techniques which aim at suggesting new items that may possibly be of interest to a user. Context-Aware Recommender Systems exploit contextual information to provide more adequate recommendations. In this paper we described a modification of an existing contextual post-filtering algorithm which uses rules-like user representation called Contextual Conditional Preferences. We extended the algorithm by taking into account rules quality measures while recommending items to a user. We proved that this modification increases the quality of recommendations, measured with *precision*, *recall* and *nDCG*, and has no impact on the execution time of the original algorithm.

I. INTRODUCTION

RECOMMENDER Systems (RS) were created as a response to the information overload problem, which we suffer from nowadays. These software tools and techniques aim at suggesting new items that may possibly be of interest to a user [1]. An item could be a movie (Netflix), a song (Pandora), a job (LinkedIn) or a friend (Facebook). In everyday life we interact with RS when we search for information using Google or when we buy something through the Internet.

Context-aware RS (CARS) are a particular category of RS which exploit contextual information to provide more adequate recommendations [2]. For example, a movie recommendation for a Saturday evening with your friends should be different from one suggested for a Sunday afternoon with your family. It has been proven that adding contextual information in the process of recommendation can highly increase prediction accuracy and user satisfaction [3]. Adomavicius and Tuzhilin [2] distinguish three main types of context-aware recommender systems, i.e. contextual pre-filtering, contextual post-filtering and contextual modeling. The paradigms differ in the way they incorporate context in the recommendation process. More details are given in Section II.

Karpus et al. [4] proposed a context-aware re-ranking algorithm (re-rankCCP) which utilizes user model called Contextual Conditional Preferences (CCPs). CCPs are special kind of rules which are learned from past user ratings and used to reorder items in a primary recommendation list. This method seems promising in making user explanations for recommendations due to the use of rules that are easy to understand by a human. However, this solution has a big disadvantage. While using CCP, an algorithm only checks its relevance to a current user context, not taking into consideration the quality of an induced preference. Thus, better CCPs can be

omitted during reordering what would lead to a reduction in the recommendation accuracy and user satisfaction.

In this paper we propose a method for determining the CCP quality using rules quality measures, i.e. *coverage*, *support* and *confidence* and apply it in the modification of the re-rankCCP. We proved that this modification increases the quality of recommendations, measured with *precision*, *recall* and *nDCG*, and has no impact on the execution time of the re-rankCCP. The main contribution of this paper can be summarized as follows:

- We propose a way to measure quality of CCP with usage of rules quality measures.
- We improve re-rankCCP algorithm to take into account quality of CCPs while generating recommendations.
- We compare effectiveness of modified re-rankCCP on 2 baseline algorithms, 3 rules quality measures and 4 aggregate functions and show that there is a best configuration for the dataset used.

The rest of the paper is organized as follows. Related work and basic re-rankCCP are described in Sections II and III, respectively. Section IV provides technical details of the proposed modification and is followed by a description of the dataset used. Section VI introduces evaluation method while obtained results are presented in Section VII. Conclusions close the paper.

II. RELATED WORK

Adomavicius and Tuzhilin [2] distinguish three main types of CARS, i.e. contextual pre-filtering, contextual post-filtering and contextual modeling. The paradigms differ in the way they incorporate context in the recommendation process.

In contextual pre-filtering, we first do selection of ratings by taking only relevant context into account. Thus, we filter an initial set of ratings and return the contextualized data. After this preparation any known two-dimensional recommendation algorithm could be used to predict user preferences. Baltrunas et al. [5] introduced *micro profiles* which split a user profile into partitions depending on the values of context parameters. They showed that usage of such *micro profiles* gave a significant improvement in the prediction accuracy in the movie domain while considering time as a context variable. Pre-filtering approach which utilizes ontological user profiles was proposed by Karpus et al. [6]. Each user profile consists of many ontologies representing user preferences in

different context and domain. Ferdousi et al. [7], [8] tried to find a correlation between ratings and context in which they were given. They proposed a new context representation based on the Pearson Correlation Coefficient as well as a new pre-filtering technique based on this representation.

Contextual post-filtering applies context after traditional recommendation process. It means that from a predicted set of recommendations we select just those that match current user context. Bahramian et al. [9] proposed a new context-aware tourism recommender system based on an ontology approach where a spreading activation technique is used to contextualize user preferences and learns the user profile dynamically. Negre et al. [10] introduced a context-aware recommender system based on a contextual post-filtering for OLAP queries, where queries recommended by a classic log-based recommender system were contextualized.

Contextual modeling differs radically from previously described paradigms. In this kind of recommenders we incorporate a context in a prediction model. The recommendations are achieved directly from the model, taking into account current user-context situation. Iqbal et al. [11] introduced Kernel Context Recommender System, which is a flexible, fast, and accurate kernel mapping framework that recognizes the importance of context and incorporates the contextual information using kernel trick while making predictions. Zheng et al. [12] proposed method that combines context-aware and multi-criteria recommender systems. They evaluated their solution on an educational data and an extended TripAdvisor dataset. Authors tested different approaches for incorporating context in the recommendation process.

In the recent years, an application of artificial neural networks in CARS is getting more and more attention [13], [14], [15]. Hildebrandt et al. [15] proposed NECTR, a novel recommender system based on a tensor factorization model and an autoencoder-like neural network. A Deep Learning based model which learns customer similarity from the sequence to sequence similarity as well as item to item similarity by considering all features of the item, contexts, and rating components was introduced by Kala et al. [14]. The method uses Dynamic Temporal Warping distance measure for dynamic temporal matching and 2D-GRU (Two Dimensional-Gated Recurrent Unit) architecture.

III. BACKGROUND - RE-RANKCCP ALGORITHM

Contextual Conditional Preferences (CCPs) were introduced to provide compact and context-aware representation of user interests for RS [16], [4]. CCP is an expression of the form:

$$(\gamma_1 = c_1) \wedge \dots \wedge (\gamma_n = c_n) \mid (\alpha_1 = a_1) \succ (\alpha_1 = a'_1) \wedge \dots \wedge (\alpha_m = a_m) \succ (\alpha_m = a'_m)$$

with γ_i being contextual variables, α_i item attributes, and $c_1, \dots, c_n, a_1, a'_1, \dots, a_m, a'_m$ being exact values of these parameters. Symbol \succ denotes a preference relation, e.g. $x \succ y$ means that someone prefers x over y .

The above CCP is read as *given the context* $(\gamma_1 = c_1) \wedge \dots \wedge (\gamma_n = c_n)$ *I prefer* a_1 *over* a'_1 *for* α_1 *and* \dots *and* a_m *over* a'_m *for* α_m . An example of the CCP is shown below.

$$\begin{aligned} \text{time of day} &= \text{afternoon} \wedge \text{companion} = \text{with children} \\ &\mid \text{genre} \in \{\text{animated}, \text{family}\} \succ \text{genre} \in \{\text{thriller}\} \end{aligned}$$

It means that for a given context, i.e. in the afternoon and the company of children, a user prefers movies that belong to the genre “animated” or “family” to those with category “thriller”.

CCPs can be learned from explicit user ratings [17]. In order to elicit preference relations the dataset containing ratings, contextual parameters and item features is split into two parts, i.e. positive and negative, based on the value of the ratings. Then, both subsets are divided into smaller sets containing all of the contextual information and one of the item features. Such prepared data are an input for the Prism[18] algorithm. Final CCPs are obtained by merging rules with the same context.

An algorithm for generating a list of top k recommendations with CCPs, the re-rankCCP, was introduced by Karpus et al. in [4]. We describe it below.

For a certain user and his current context, first we generate a primary list of top m recommendations with some existing non-context-aware algorithm, e.g., UserKNN. The value of m has to be significantly greater than k , where k is the number of the recommendations in the final list. Then we have to find the best CCPs that will be further used in the reshuffling process.

The best CCPs are those which are most similar to the considered context. In order to count a contextual similarity between a CCP p and a current user context $ctx(u)$ we used the following measure:

$$\text{sim}(p, ctx(u)) = \sum_{(\gamma_i, c_i) \in p} \text{overlap}(ctx(u), (\gamma_i, c_i)) \quad (1)$$

We also used the overlap function defined as:

$$\text{overlap}(ctx(u), (\gamma_i, c_i)) = \begin{cases} 1 & (\gamma_i, c_i) \in ctx(u); \\ 0.5 & c_i = -1; \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

The overlap function returns 1 when we are sure that the pair (γ_i, c_i) is contained both in the contextual part of p and in the current user context $ctx(u)$. When it is uncertain, i.e. when the value c_i for the dimension γ_i is equal to -1 (the unknown value), it returns 0.5. Otherwise 0 is returned. Note that the current user context $ctx(u)$ is also a set of pairs (γ'_i, c'_i) , i.e. the name of the contextual variable and its value.

For each item in the primary recommendations list and each best CCP we have to compute how much an item i satisfies a CCP p . For this purpose, we have to use the *satisfiability* measure:

$$\text{sat}(i, p) = \frac{\sum_{\alpha \in a(p)} (\text{sim}(v_\alpha^m(p), v_\alpha(i)) - \text{sim}(v_\alpha^l(p), v_\alpha(i)))}{|a(p)|},$$

where sim denotes Jaccard similarity, α is the name of an item feature, $a(p)$ is the set of item attributes considered in

the CCP p , $v_\alpha(i)$ is the set of values of an attribute α for an item i . Similarly $v_\alpha^m(p)$ and $v_\alpha^l(p)$ denote the sets of values of an attribute α for a CCP p on both sides of the preference relation - m stands for *more preferred* and l for *less preferred*.

The *satisfiability* measure represents the difference between item similarities to the both sides of the CCP preference relation, i.e. the similarity to the most preferred part minus the similarity to the less preferred part. In this way we reward items that fit best to user preferences and penalize items that have features that user does not like. The size of a set of item attributes serves as a normalization factor. Thus, regardless of the number of item features, the value of *satisfiability* is always between 0 and 1.

The next step is to order the primary recommendations list according to the value of average *satisfiability* of the best CCPs. The last part is to cut off unneeded items from resulting recommendations list to receive top 5, top 10 or other top k ranking.

IV. ALGORITHM MODIFICATION

One of the key parts of the re-rankCCP algorithm is a selection of best CCPs for current user context based on the similarity measure from Equation 1. However, this measure does not take into account the quality of CCPs. Consequently, recommendation could be made based on less important user preferences. Therefore, we replaced the similarity with a weighted similarity sim_w :

$$sim_w(p, ctx(u)) = q(p)sim(p, ctx(u)) ,$$

where sim is the similarity measure from Equation 1 and $q(p)$ is a quality of a CCP p . Now, we need to define the quality of a CCP.

CCPs can be generated from rules induced with Prism algorithm. Thus, we decided to use rules quality measures, like *coverage* or *support*, to define quality of CCP. However, one CCP is created using many different rules. Therefore, we have to decide how to reasonably aggregate many rules quality values into one value characterizing CCP's quality.

In this paper we tested four aggregate functions, which we found the most reasonable, i.e. *minimum*, *maximum*, *sum* and *average*. The last one seems the most obvious because it simply takes quality values from all used rules and returns one normalized value for a CCP. We also obtain standardized quality for the first two functions which simply take the worst and the best rule quality value, respectively. The *sum* function additionally reflects the quantity of rules that are used for creation of a CCP. The more good rules were used, the higher the quality of a CCP would be.

We decided to apply three commonly used rules quality measures, namely: *coverage*, *support* and *confidence* [19]. For this purpose, we had to slightly modify an algorithm for CCPs extraction to compute those measures. However, this algorithm is independent from the re-rankCCP. We also extended a CCP representation to contain information about its quality. Figure 1 shows modified CCP from the above example in the JSON format. The rest of the re-rankCCP remains the same.

```

1  {"CCPs":[
2    { "context":{"time of day":"afternoon", "companion":"with children"},
3      "MorePreferred":{"genre":["animated","family"]},
4      "lessPreferred":{"genre":["thriller"]},
5      "coverage":0.4924},
6    ...
7  ]}
```

Fig. 1. An example CCP in the JSON format with information about rule quality measured with the coverage.

V. DATASET

We performed experiments on the same dataset as authors of the original re-ranking algorithm, i.e. LDOS-CoMoDa dataset. The LDOS-CoMoDa dataset [20] was collected by a web application that enables contextual rating of a movie just after watching it. The dataset consists of 2296 ratings given by 121 users to 1232 items. It contains 30 variables among which 12 are contextual parameters. Other variables are basic information about user (user id, age, sex, city and country), a rating in a 5-star scale (higher values denote higher preference) and content information about multiple item dimensions (item id, director, country, language, year, 3 main genres, 3 main actors and budget). Unknown values are denoted by “-1”.

We chose users who rated at least 5 items. Then, we randomly selected 20% of items rated by each of these users to be included in the test set. The remaining data constitute the training set.

VI. EVALUATION METHOD

We re-implemented in Python the re-rankCCP algorithm, which was originally implemented in Java. We also performed new training and test sets split. Thus, because of the randomness of the split, we have different data in those sets than in the previous papers [4], [16].

The re-rankCCP is a post-filtering technique which means that it needs other algorithm to work. We decided to test two known methods, i.e. Bayesian Personalized Ranking (BPR)[21] and User k Nearest Neighbors (UserKNN)[22]. We had several reasons for this choice. First of all, inventors of re-rankCCP obtained the most promising results with BPR algorithm. Second of all, UserKNN was one of the most (or even the most) popular method in the field of RS. Last but not least is the way how these algorithms treat missing data. BPR tries to minimize its negative impact on the prediction accuracy, while UserKNN completely ignores missing data. Hereby, we obtained a representative sample of base algorithms.

For the re-rankCCP and base algorithms we had to set up some parameters. UserKNN used 50 neighbors to compute recommendations. Base algorithms generate lists of top 100 items while re-rankCCP produces the top 10 list. Rating greater than 3 is considered positive. We decided to choose three commonly used measures of recommendations quality, i.e. *precision*, *recall* and *nDCG* [23].

In addition to the impact of the modification on the quality of recommendations, we wanted to check its impact on the algorithm execution time. Since we slightly modified CCP representation, our method does not affect the time

of generating recommendations. However, it could have impact on the time needed to induce CCPs, since it is where the CCP quality is computed. In order to check it we performed an experiment for which we prepared 4 datasets from the LDOS-CoMoDa. The datasets consists of 3000, 5000, 7000 and 10000 rows respectively. For each dataset we performed CCPs generation for the re-rankCCP algorithm and its modifications. We collected results with *%time* function which is available in *ipython* environment.

VII. RESULTS

Table I shows values of *precision*, *recall* and *nDCG* obtained for different configurations of algorithms, rules quality measures and aggregate functions during our experiments. The best results for each algorithm and rules quality measure is marked with bold (locally best result), while the best results for each algorithm/base algorithm is marked with underline (globally best result considering division into two groups: BPR and UserKNN). It should be noticed that re-rankCCP always improves *precision*, *recall* and *nDCG* of its base algorithm. Nonetheless, re-rankCCP performs weaker than its modifications with rules quality measures.

For most of configurations of algorithms and rules quality measures, the best results were obtained by *minimum* and *average* functions. The first function was the best for *support* and *confidence*, irrespective of a base algorithm, while the latter works well with *coverage* and *support* on re-rankCCP with BPR algorithm. An exception appears in re-rankCCP with UserKNN algorithm and *coverage* measure. The best results for this configuration was obtained by the *maximum* function. The best results for *minimum* and *average* functions should not be surprising. While using *minimum* function, we assure that all other rules used to induce a CCP have greater quality values than the resulting value. We obtain similar effect for the *average*. On the contrary, for the *maximum* we could choose preference which is generated from rules from which one is strong and all others are weak. The same bad effect can happen for *sum* function. Modified re-rankCCP will prefer a CCP from many weak rules than a CCP from two strong rules. The smallest improvement in modified re-rankCCP was obtained with the *coverage* for both base algorithms. It can be justified by the fact that the *coverage* is not a proper quality measure for rules since it considers only antecedent of a rule.

To check the statistical significance of obtained results we performed Wilcoxon signed rank test with $\alpha = 0.05$. For the re-rankCCP with BPR as a baseline algorithm two results were statistically insignificant, i.e. for *coverage-minimum* and *confidence-maximum* pairs with p-value equals to 0.4375 and 0.5282, respectively. The re-rankCCP with UserKNN as a baseline and *support* measure has almost all results insignificant, i.e. for *maximum*, *average* and *sum* functions with p-values equal to 0.5745, 0.0625 and 0.1563 respectively. All other reported results are statistically significant.

Considering results presented in Table I and their statistical significance, we can conclude that objectively the best improvement to the re-rankCCP is obtained using the *support*

measure for rules quality and the *minimum* function for an aggregation. It should be noticed that UserKNN performs pretty weak on LDOS-CoMoDa dataset. This could be because of the data sparsity.

Table II shows times of generating CCPs for the re-rankCCP algorithm and its modification with the coverage. We obtained very similar results for support and confidence measures which is why we omitted it here. The differences in execution times are negligible. Thus, we can conclude that our modification does not increase execution time of re-rankCCP, and improves the quality of recommendations.

VIII. CONCLUSIONS

In this paper we proposed a way for measuring the CCP quality using rules quality measures and aggregate functions. To the best of our knowledge, this is the first attempt to compute a CCP quality. We also improved re-rankCCP algorithm by incorporating quality of CCPs into recommendation process and proved that this modification outperforms the re-rankCCP as well as both baseline algorithms, i.e. BPR and UserKNN. We compared its effectiveness on two baseline algorithms, three rules quality measures, i.e. *coverage*, *support* and *confidence*, and four aggregate functions, i.e. *minimum*, *maximum*, *sum* and *average*. Our experiments showed that the *support* measure aggregated with the *minimum* function is the best configuration for computing the CCP quality on LDOS-CoMoDa dataset. However, more experiments on other datasets and with more baseline algorithms are needed to check if these results could be generalized.

REFERENCES

- [1] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich, *Recommender Systems: An Introduction*, 1st ed. New York, NY, USA: Cambridge University Press, 2010.
- [2] G. Adomavicius and A. Tuzhilin, in *Handbook on Recommender Systems*, S. B. Ricci F., Rokach L. and K. P. B., Eds. Springer, 2011, ch. Context-Aware Recommender Systems, pp. 217–256.
- [3] M. Kristoffersen, S. Shepstone, and Z.-H. Tan, "The importance of context when recommending tv content: Dataset and algorithms," *IEEE Transactions on Multimedia*, vol. 22, no. 6, pp. 1531–1541, 2020.
- [4] A. Karpus, T. di Noia, and K. Goczyła, "Top k recommendations using contextual conditional preferences model," in *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems, FedCSIS 2017, Prague, Czech Republic, September 3-6, 2017.*, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., 2017, pp. 19–28. [Online]. Available: <https://doi.org/10.15439/2017F258>
- [5] L. Baltrunas and X. Amatriain, "Towards time-dependant recommendation based on implicit feedback," in *Proceedings of 1st Workshop on Context-Aware Recommender Systems*, 2009.
- [6] A. Karpus, I. Vagliano, and K. Goczyła, "Serendipitous recommendations through ontology-based contextual pre-filtering," *Communications in Computer and Information Science*, vol. 716, pp. 246–259, 2017.
- [7] Z. V. Ferdousi, D. Colazzo, and E. Negre, "Correlation-based pre-filtering for context-aware recommendation," in *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, March 2018, pp. 89–94.
- [8] Z. V. Ferdousi, D. Colazzo, and E. Negre, "Cbpf: Leveraging context and content information for better recommendations," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11323 LNAI, pp. 381–391, 2018.
- [9] Z. Bahramian, R. Abbaspour, and C. Claramunt, "A context-aware tourism recommender system based on a spreading activation method," K. F. Samadzadegan F., Ed., vol. 42, no. 4W4. International Society for Photogrammetry and Remote Sensing, 2017, pp. 333–339.

TABLE I
RESULTS OBTAINED FOR DIFFERENT ALGORITHMS, RULES QUALITY MEASURES AND AGGREGATE FUNCTIONS.

Algorithm	Base algorithm	Rules quality measure	Aggregate function	Precision	Recall	nDCG
re-rankCCP	BPR	coverage	maximum	0.06043	0.03444	0.19368
			minimum	0.05468	0.03273	0.19309
			average	0.07098	0.04173	0.20135
			sum	0.05875	0.03922	0.18690
		support	maximum	0.08129	0.03909	0.27293
			minimum	0.10408	0.04610	0.30538
			average	0.09185	0.04705	0.30710
			sum	0.08417	0.04142	0.25481
		confidence	maximum	0.05707	0.03022	0.20156
			minimum	0.08729	0.04037	0.26302
			average	0.07266	0.03042	0.23508
			sum	0.07242	0.03370	0.22983
re-rankCCP	BPR		0.05366	0.02668	0.18045	
	BPR		0.04508	0.02323	0.17693	
re-rankCCP	UserKNN	coverage	maximum	0.01847	0.01161	0.09903
			minimum	0.01703	0.01004	0.08884
			average	0.01583	0.00965	0.08128
			sum	0.01703	0.01004	0.08826
		support	maximum	0.02038	0.00856	0.08085
			minimum	0.03381	0.01290	0.17228
			average	0.01894	0.00797	0.08139
			sum	0.02062	0.00830	0.08924
		confidence	maximum	0.02662	0.01085	0.12977
			minimum	0.03237	0.01238	0.14131
			average	0.02542	0.01032	0.12725
			sum	0.02758	0.01076	0.13002
re-rankCCP	UserKNN		0.01570	0.00849	0.07800	
	UserKNN		0.01103	0.00719	0.03580	

TABLE II
TIMES OF GENERATING CCPs FOR THE RE-RANKCCP ALGORITHM AND ITS MODIFICATION WITH COVERAGE.

Number of rows	Execution time	
	With coverage	Without coverage
3000	26.5 s	26.5 s
5000	41.1 s	41.2 s
7000	59.6 s	59.7 s
10000	1min 24s	1min 24s

[10] E. Negre, F. Ravat, and O. Teste, "Olap queries context-aware recommender system," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11030 LNCS, pp. 127–137, 2018.

[11] M. Iqbal, M. Ghazanfar, A. Sattar, M. Maqsood, S. Khan, I. Mehmood, and S. Baik, "Kernel context recommender system (kcr): A scalable context-aware recommender system algorithm," *IEEE Access*, vol. 7, pp. 24 719–24 737, 2019.

[12] Y. Zheng, S. Shekhar, A. A. Jose, and S. K. Rai, "Integrating context-awareness and multi-criteria decision making in educational learning," in *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, ser. SAC '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 2453–2460.

[13] J. Manotumrukha, "Deep collaborative filtering approaches for context-aware venue recommendation," in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 1383. [Online]. Available: <https://doi.org/10.1145/3077136.3084159>

[14] K. Kala and M. Nandhini, "Gated recurrent unit architecture for context-aware recommendations with improved similarity measures," *KSII Transactions on Internet and Information Systems*, vol. 14, no. 2, pp. 538–561, 2020.

[15] M. Hildebrandt, S. Sunder, S. Mogoreanu, M. Joblin, A. Mehta, I. Thon, and V. Tresp, "A recommender system for complex real-world applications with nonlinear dependencies and knowledge graph context," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11503 LNCS, pp. 179–193, 2019.

[16] A. Karpus, T. di Noia, P. Tomeo, and K. Goczyla, "Rating prediction with contextual conditional preferences," in *Proceedings of the 8th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2016) - Volume 1: KDIR, Porto - Portugal, November 9 - 11, 2016*, A. L. N. Fred, J. L. G. Dietz, D. Aveiro, K. Liu, J. Bernardino, and J. Filipe, Eds. SciTePress, 2016, pp. 419–424. [Online]. Available: <http://dx.doi.org/10.5220/0006083904190424>

[17] A. Karpus, "Context-aware user modelling and generation of recommendations in recommender systems," Ph.D. dissertation, Gdańsk University of Technology, 2018.

[18] J. Cendrowska, "PRISM: an algorithm for inducing modular rules," *International Journal of Man-Machine Studies*, vol. 27, no. 4, pp. 349–370, 1987.

[19] J. M. Luna, M. Ondra, H. M. Fardoun, and S. Ventura, "Optimization of quality measures in association rule mining: an empirical study," *International Journal of Computational Intelligence Systems*, vol. 12, pp. 59–78, 2018. [Online]. Available: <https://doi.org/10.2991/ijcis.2018.25905182>

[20] A. Kosir, A. Odic, M. Kunaver, M. Tkalcic, and J. F. Tasic, "Database for contextual personalization," *Elektrotehniški vestnik [English print ed.]*, vol. 78, no. 5, pp. 270–274, 2011.

[21] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "Bpr: Bayesian personalized ranking from implicit feedback," in *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, ser. UAI '09. Arlington, Virginia, United States: AUAI Press, 2009, pp. 452–461. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1795114.1795167>

[22] J. A. Konstan, B. N. Miller, D. Maltz, J. L. Herlocker, L. R. Gordon, and J. Riedl, "GroupLens: Applying collaborative filtering to usenet news," *Commun. ACM*, vol. 40, no. 3, pp. 77–87, Mar. 1997. [Online]. Available: <http://doi.acm.org/10.1145/245108.245126>

[23] G. Shani and A. Gunawardana, "Handbook on recommender systems," S. B. K. P. B. Ricci F., Rokach L., Ed. Springer, 2011, ch. Evaluating Recommendation Systems, pp. 257–298.

Using Transformer models for gender attribution in Polish

Karol Kaczmarek
 Adam Mickiewicz University,
 Faculty of Mathematics and Computer Science,

Applica.ai Sp. z o.o.
 Email: karol.kaczmarek@amu.edu.pl

Jakub Pokrywka, Filip Graliński
 Adam Mickiewicz University,
 Faculty of Mathematics and Computer Science,
 Uniwersytetu Poznańskiego 4,
 61-614 Poznań, Poland
 Email: {firstname.lastname}@amu.edu.pl

Abstract—Gender identification is the task of predicting the gender of an author of a given text. Some languages, including Polish, exhibit gender-revealing syntactic expression. In this paper, we investigate machine learning methods for gender identification in Polish. For the evaluation, we use large (780M words) corpus "He Said She Said", created by grepping (for author's gender identification) gender-revealing syntactic expressions and normalizing all these expressions to masculine form (for preventing classifiers from using syntactic features). In this work, we evaluate TF-IDF based, fastText, LSTM and RoBERTa models, differentiating self-contained and non-self-contained approaches. We also provide a human baseline. We report large improvements using pre-trained RoBERTa models and discuss the possible contamination of test data for the best pre-trained model.

I. INTRODUCTION

The task of *gender identification or attribution* consists in predicting the gender of an author of a given text. As such, it is an example of text classification, is usually tackled using supervised machine learning, and is relatively popular in the NLP community. Some recent example of experiments in automatic gender identification for various languages are: [17], [27], [2], [14]. For a critical analysis of gender detection systems and their limitations, see [18].

Collections of gender-labeled texts are required if a system based on supervised machine learning is to be trained. The usual approach is to use metadata such as information on authors (of books, papers, social media posts, etc.). Interestingly, some languages exhibit gender-revealing first-person expressions (cf. *soy polaco* vs *soy polaca* in Spanish), and such expressions can be used to automatically label texts as written by a male or female in order to create a data set. This approach (*distant supervised learning*, [21]) is similar to using emoticons for sentiment analysis tasks [23], [9].

Some languages (e.g. Slavic languages) are more amenable to this distant supervised approach than others (e.g. English or Chinese). The approach was applied to Polish to create a large collection of texts, the "He Said She Said" (HSSS) corpus [10]. In this paper, we (1) re-state the original challenge as a classification task with a probability-based evaluation metric, (2) report on large improvements on the gender detection task using pre-trained RoBERTa models, and (3) discuss the

TABLE I
 THE "HE SAID SHE SAID" CHALLENGE IN NUMBERS.

		characters	words	items
train	total	1,240,131,217	177,428,897	3,601,424
	male	628,793,876	89,795,752	1,800,712
	female	611,337,341	87,633,145	1,800,712
dev-0	total	51,080,450	7,158,683	137,314
	male	26,066,897	3,641,716	68,657
	female	25,013,553	3,516,967	68,657
dev-1	total	51,009,045	7,275,691	156,606
	male	25,579,703	3,641,568	78,303
	female	25,429,342	3,634,123	78,303
test-A	total	43,597,629	6,234,069	134,618
	male	22,253,841	3,175,881	67,309
	female	21,343,788	3,058,188	67,309

possible contamination of test data with the data on which RoBERTa models were trained.

In Section II, we discuss the HSSS challenge along with the modifications in the data set done for the purposes of this paper. In the main Section III, we discuss the methods we applied to tackle the challenge of gender identification. Section IV summarizes the results. Finally, we discuss the issues of training/testing data contamination in Section V.

II. HE SAID SHE SAID TASK

Polish is one of the languages with a high frequency of gender-specific first-person expressions. (Only the few languages with gender distinction in the first person, e.g. Ngala [24], might have a higher frequency of such expressions.) This fact was leveraged to create a large gender-labeled corpus for Polish: the "He Said She Said" corpus [10]. Simply CommonCrawl dataset was grepped, using morphological dictionaries and handcrafted rules, for gender-specific first-person expressions. Obviously, there were some issues that needed to be addressed, e.g. quotes, titles, SEO spam.

Later, the corpus was turned into a classification challenge hosted at the Gonito.net platform [11]. All feminine gender-

specific first-person expressions were changed to masculine forms in order to prevent classifiers from using the simple gender-revealing syntactic features. Obviously, without this normalization step, the challenge would be trivial. The corpus was randomly split into 4 sets: train set, two development (validation) sets (dev-0 and dev-1) and test set (test-A). The split was based on the websites from which the texts originated, i.e. texts from the same website would belong to the same set. Also, the sets were balanced so that 50%/50% distribution would be obtained, not just for the whole data set, but also for *each* website. For instance, let's consider a message board about pregnancy, in general, there are many more texts written by women there (at least judging by gender-marked first-person expressions), but for the challenge, the same number of male and female texts would be sampled from such a website. This, along with the fact that texts are short, makes the challenge rather difficult.

The challenge was presented [11] to showcase the Gono.net platform and was discussed there only briefly. For more detailed information about the challenge, see Table I.

For this paper, two changes have been made to the original challenge:

- 1) *Likelihood* metric was chosen as the main metric (instead of simple accuracy), Likelihood is defined as the geometric mean of probabilities assigned to the gold-standard classes – the motivation was that accuracy is not enough to distinguish solutions of varying quality and confidence;
- 2) some unwanted blank characters were removed.

Some initial experiments with learning classifiers based on the HSSS data set were presented in [12].

III. METHODS

We introduce the structure of our experiments as follows. Subsection III-A describes human baselines. Subsection III-B describes TF-IDF (term frequency-inverse document frequency) based methods. Subsection III-C describes some neural methods. Both III-B and III-C are self-contained. This means not including any data apart from training data available in HSSS task. Subsection III-D describes pre-trained transformer models. Table III presents all classifiers results.

- self-contained – we use only data available from the HSSS task: train on the training set, validate on the dev-0 (validation) set and report results on the test-A (test) set. We will use 256 sequence length which covers most (over 90%) of the HSSS data to speed up the training process.
- non-self-contained – we use publicly available models, which were pre-trained on large amounts of data (may be contaminated by examples from the test or validation set). We will use the sequence length that was saved for these models, which is usually 512.

TABLE II
RESULTS ON THE TEST SET SAMPLE OF SIZE 800 CREATED FOR HUMAN EVALUATION.

method	test accuracy
TF-IDF + logistic regression	0.68500
Polish RoBERTa base	0.77125
LSTM (constrained)	0.73375
human 1	0.65250
human 2	0.67375
human 3	0.66250
human 4	0.65625
human ensemble	0.68125

A. Human Baseline

Four people (two females and two males) made predictions for random sample sets of size 200 for development set and 800 for the test set. They were explained how the dataset was created and asked not to look for the answer on the internet. We rejected human 1 result based on the development dataset result and created a human ensemble with the remaining 3 people predictions using majority voting. The results are presented with the best TF-IDF based, self-contained and overall methods in the Table II.

B. TF-IDF based methods

Term frequency-inverse document frequency (TF-IDF) is a common vector representation of a document in natural language processing. We use the TfidfVectorizer library from Scikit-learn with standard parameters. This includes word-level, lowercasing, l_2 normalization. We did not restrict the vocabulary size and we used word-level splitting. The following classifiers were trained using TF-IDF vectors: Logistic Regression, XGBoost Classifier, SVM.

1) *Logistic Regression*: We used LogisticRegression from Scikit-learn library with standard parameters, except for the maximum number of iteration. We trained until classifier convergence.

2) *Support Vector Machine Classifier*: Support-Vector Network [5] is a common algorithm, that circumvents non-linear separability of data as well as separate samples from different categories. Although, in this case, we chose LinearSVC from Scikit-Learn, which uses a linear kernel. The reason is memory and computation issues related to the high dimension of TF-IDF representation and the number of samples in the HSSS task. Again, we used standard parameters, except for no maximum number of iteration, which led to convergence. We do not report likelihood due to the fact that SVM does not yield probabilities.

3) *XGBoost Classifier*: Tree boosting is an effective and popular method for regression and classification. We used XGboost library [3] with the choice of the parameters suited for better classifier quality.¹ This includes gbtrees booster,

¹Some of the parameters were taken from <https://www.kaggle.com/serigne/stacked-regressions-top-4-on-leaderboard>

learning rate set to 0.05 and max depth set to 3.

C. Neural Methods (*self-contained*)

1) *FastText*: FastText [15] is a shallow neural network library created for fast text classification model training and evaluation. We used a supervised setting with hyperparameter tuning, the word embeddings were initialized randomly. The best result was obtained with wordNgrams set to 2, word dimension set to 156, and context size window set to 5.

2) *LSTM*: Long Short Term Memory Networks [13] were used to obtain a state-of-the-art results on most NLP tasks before the era of Transformer language models [7]. In our tasks, for bidirectional LSTM, SentencePiece [19] tokenization performs better than word-level lowercase tokenization. Vocab size 50k was used with randomly initialized embeddings of size 100. We tried embedding size 300, but resulted in slightly worse classifier quality. We used one layer of 256 units, trained with Adam [16] optimizer with learning rate 0.001. The batch size used for training was 400 and sequences were trimmed and padded to 256 tokens.

3) *Transformer*: In the last time Transformer [26] and its modification like BERT [7], RoBERTa [20] or XLM-R [4] achieve state-of-the-art in the benchmarks such as GLUE [29] or SuperGLUE [28] benchmark. Most often used bidirectional Transformers are pre-trained on huge amounts of monolingual data in the Masked Language Model (MLM) process, where the model learns a bidirectional representation of tokens. Next, pre-trained models are finetuned to the specific task. This process reduces the time to train a new model from scratch and can be easily adapted to other tasks. In our case, the downstream task is classification, where the model uses a special token ([CLS], classification token), which represents the whole sentence and helps achieve better results.

We train self-contained classifier based on the RoBERTa model in two ways: with pre-training and without pre-training (train classifier from the scratch) stage. We only used the data that was available in the HSSS challenge to avoid any data leaks in the other data sets. To compare our methods we created Transformer with 8 layers, 8 heads, 256 sequence length and embedding size 512 and 2048 respectively for internal model representation and feed forward layer (after attention layer). We use 50k size vocabulary with Sentencepiece tokenization and randomly initialized embeddings of size 512. First, the model was pre-trained for 10 epochs with Masked Language Model (MLM) criterion and finetuned 10 epochs for the classification tasks. Second, the model was trained on the classification task for 20 epochs (comparing to the previous one, where it was 10 + 10 epochs for pre-training and classification) only. We pre-train and finetune with Adam optimizer with learning rate 0.0001 and 50 sentences per batch. Scores presented in the Table III show that the pre-training stage is the important element to achieve a better model for classification tasks.

D. Pre-trained Transformers

In this section we describe fine-tuning of models publicly available for Polish language: Polish RoBERTa [6] and multi-

lingual XLM-R [4] (which supports 100 languages including Polish). Both models are available in the two versions: base (with 12 layers) and large (with 24 layers). Monolingual models like RoBERTa are focused on achieving the best results in a given language. On the other hand, multilingual models support as many languages as possible with results similar to monolingual models. The disadvantage of multilingual models is the size of the vocabulary, which is several times larger than monolingual models like Polish RoBERTa. Bigger vocabulary needs more resources to fine-tune models, but may improve results by cross-language relationships.

1) *Polish RoBERTa finetuning*: We finetuned Polish RoBERTa [6] (base and large model) using fairseq library [22] for 5 and 3 epochs respectively for the base and large model. Further training resulted in lower development dataset accuracy. We used Adam optimizer with a learning rate 0.00001 and around 200k warmup steps. The maximum sequence we use is 512 as in original Polish RoBERTa.

2) *Polish RoBERTa finetuning with Monte-Carlo model averaging*: Common practice when using dropout is to scale weights during inference time. However, as described in [25] (section 7.5), further investigated in [8], this procedure is only an approximation of Monte-Carlo model averaging. We checked, whether the Monte-Carlo model averaging yields better results than standard weight scaling in our case. By setting Polish RoBERTa (both base and large) in the training mode (with active dropout), making predictions 12 times, and averaging likelihood, we obtained slightly better results in both cases.

3) *XLM-R finetuning*: We finetuned multilingual XLM-R [4] base and large for 1 epoch, further training does not improve results. Each of the models was trained with 512 tokens using Adam optimizer with a learning rate 0.00004. Batch size has been set to 10 and 25 for the base and large model. Results are available in the Table III.

4) *Polish RoBERTa last layer averaged*: For the evaluation of how much information about language Polish RoBERTa possesses, we conducted the following experiment. We extracted the last layer tokens and averaged them. Then, we trained logistic regression classifier with no Polish RoBERTa finetuning. This was done until classifier convergence.

5) *XLM-R last layer averaged*: We conducted the same experiment with XLM-R as in subsection III-D4.

6) *Polish RoBERTa fill mask*: In order to check the predicting power of only pre-trained Polish RoBERTa models, we conducted the following experiment. We masked all gender-revealing first-person expression and used the models in Masked Language Model setting. We choose one random expression and looked for the most probable word indicating gender in the first 10 model predictions. Only 6333 samples out of 137314 in the test set did not reveal first-person expression in the first 10 predictions. No training or development sets were used in this experiment. However, this method does not yield good results (though the trivial baseline was beaten).

IV. RESULTS

The self-contained models (BiLSTM and RoBERTa MLM + classifier) achieved better results than TF-IDF and fastText. The BiLSTM model achieves a bit better results than the Transformer base model, which suggests that the Transformer model needs more resources. The classifier trained from scratch (without pre-training) produces inferior results, and this shows again that the pre-training step is an important element in classification tasks. Neural methods achieve better results than the human baseline, but human results are comparable to TF-IDF.

Pre-trained models trained on the much larger data set than the HSSS data set achieve the best results. Monolingual and multilingual models achieve similar results, but XLM-R large achieve lower results than other pre-trained models, indicating that the bigger models may not improve results on the classification tasks. Polish RoBERTa large achieved similar results to the base version, which might mean that RoBERTa large needs more pre-training steps to get better results.

V. CONTAMINATION STUDY

Using a pre-trained language model (or any other solution not constrained to the train set provided with the challenge) raises the question of data contamination or train-test overlap, i.e. (1) was the test set represented in the training set of the language model?, (2) did it make the results better (e.g. due to memorization of test texts by the language model)? See [1] for the discussion of data contamination in the case of the GPT-3 model when used for popular English NLP test sets.

We carried out a contamination study on the solution based on the Polish RoBERTa model (the best solution so far). As the Polish RoBERTa was trained (among other sources) on CommonCrawl 2019/2020 [6], and the HSSS was prepared using CommonCrawl 2012-2015 (mostly 2012), the risk of contamination was real (a significant percentage of Web content from 2012-2015 could survive up to 2019).

We searched the contents of CommonCrawl 2019 (as provided to us by the authors of [6]²) for the six-gram fragments of the HSSS test set, obviously taking into account the fact that feminine gender-specific forms were modified during the preparation of the HSSS test set.

The summary of the contamination study is given in Table IV, where the results obtained with Polish RoBERTa are compared against the best constrained solution (an LSTM trained on the HSSS training set). The following conclusions can be made:

- results on the contaminated subset *are* better (and the difference of the Accuracy/Likelihood metrics on the contamination and not contaminated metric is significant), and this might indicate that the problem is real;
- still, the percentage of data contaminated is low (3%), hence the impact on the total is limited; if we were to lower the results on the contaminated subset to be

²Unfortunately, we were unable to check the other sources, though the probability of them contaminating the test set seems much lower

the same as on the uncontaminated subset, the accuracy would be lower only by a small margin;

- note that this is not a proof of contamination; the cause of better results on the contaminated subset might be different, for example it might have been caused by the fact that CommonCrawl 2019 for Polish RoBERTa was filtered by a language model, whereas for the HSSS data set — only using handcrafted heuristics, i.e. sentences might be longer and “proper” (e.g. say with fewer spam texts), hence easier for a classification task.

VI. CONCLUSIONS

We showed that a pre-trained Transformer model can obtain strong results for a challenging classification tasks on short texts. It turned out that predictions done by humans (even aggregated) were much worse. What is important is that influence of contamination of the training set was practically excluded.

REFERENCES

- [1] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.
- [2] B. Bsir and M. Zrigui. Bidirectional LSTM for author gender identification. In *International Conference on Computational Collective Intelligence*, pages 393–402. Springer, 2018.
- [3] T. Chen and C. Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, pages 785–794, New York, NY, USA, 2016. ACM.
- [4] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*, 2019.
- [5] C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.
- [6] S. Dadas, M. Perełkiewicz, and R. Poświęta. Pre-training Polish Transformer-Based language models at scale. In L. Rutkowski, R. Scherer, M. Korytkowski, W. Pedrycz, R. Tadeusiewicz, and J. M. Zurada, editors, *Artificial Intelligence and Soft Computing*, pages 301–314, Cham, 2020. Springer International Publishing.
- [7] J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR*, abs/1810.04805, 2018.
- [8] Y. Gal and Z. Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. *Proceedings of The 33rd International Conference on Machine Learning*, 06 2015.
- [9] A. Go, R. Bhayani, and L. Huang. Twitter sentiment classification using distant supervision, 2009.
- [10] F. Galiński, Ł. Borchmann, and P. Wierchoń. “He Said She Said” — a male/female corpus of Polish. In N. Calzolari, K. Choukri, T. Declerck, S. Goggi, M. Grobelnik, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, and S. Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, 2016. European Language Resources Association (ELRA).
- [11] F. Galiński, R. Jaworski, Ł. Borchmann, and P. Wierchoń. Gonito.net — open platform for research competition, cooperation and reproducibility. In A. Branco, N. Calzolari, and K. Choukri, editors, *Proceedings of the 4REAL Workshop*, pages 13–20, 2016.
- [12] F. Galiński, R. Jaworski, Ł. Borchmann, and P. Wierchoń. Vive la petite différence! Exploiting small differences for gender attribution of short texts. *Lecture Notes in Artificial Intelligence*, 9924:54–61, 2016.
- [13] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, Nov. 1997.

TABLE III

RESULTS. *HUMAN BASELINE WAS EVALUATED ONLY ON THE RANDOM SAMPLE OF SIZE 800. †REFERENCES TO REPOSITORIES AT GONITO.NET [11] ARE GIVEN IN CURLY BRACKETS. SUCH A REPOSITORY MAY BE ALSO ACCESSED BY GOING TO HTTP://GONITO.NET/Q AND ENTERING THE CODE THERE.

method	test likelihood	test accuracy	gonito submission†
human baseline*	0.00000	0.68125	{87a138}
TF-IDF + logistic regression	0.55278	0.67175	{ecc1ee}
TF-IDF + linear SVM	0.00000	0.66477	{da348f}
TF-IDF + XGBClassifier	0.54269	0.65112	{5a17c9}
fastText	0.54541	0.67448	{4d18c0}
Bi-LSTM	0.57177	0.69786	{a0d38c}
RoBERTa MLM + classifier	0.57068	0.69153	{203325}
RoBERTa classifier (only)	0.55784	0.67951	{6756e6}
Polish RoBERTa (base) finetuned	0.60913	0.74185	{049966}
Polish RoBERTa (large) finetuned	0.60503	0.74388	{2b8541}
XLM-R (base) finetuned	0.60015	0.72356	{bdac6e}
XLM-R (large) finetuned	0.57141	0.69047	{bdac6e}
Polish RoBERTa (base) active dropout	0.62110	0.74332	{ea4b15}
Polish RoBERTa (large) active dropout	0.61949	0.74406	{2e89da}
Polish RoBERTa (large) last layer + logic regression	0.54113	0.65956	{582542}
XLM-R (large) last layer + logic regression	0.54067	0.65545	{115246}
Polish RoBERTa (large) fill mask	0.00000	0.55828	{11633b}

TABLE IV

CONTAMINATED VS NOT CONTAMINATED SUBSET OF THE TEST SET. P-VALUES ARE CALCULATED WITH THE MANN–WHITNEY U TEST.

		contaminated	not-contaminated	all	p-value
items	#	4,076	130,542	134,618	
	%	3.0%	97.0%	100.0%	
Polish RoBERTa base	Likelihood	0.64656	0.62032	0.62110	0.0000
	Accuracy	0.77159	0.74244	0.74332	0.0007
LSTM (constrained)	Likelihood	0.58305	0.57142	0.57177	0.0000
	Accuracy	0.70118	0.69776	0.69786	0.3549

- [14] S. Hussein, M. Farouk, and E. Hemayed. Gender identification of Egyptian dialect in Twitter. *Egyptian Informatics Journal*, 20(2):109–116, 2019.
- [15] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*, 2016.
- [16] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 12 2014.
- [17] D. Kodyan, F. Hardegger, S. Neuhaus, and M. Cieliebak. Author Profiling with bidirectional RNNs using Attention with GRUs: Notebook for PAN at CLEF 2017. In *CLEF 2017 Evaluation Labs and Workshop—Working Notes Papers, Dublin, Ireland, 11-14 September 2017*, volume 1866. RWTH Aachen, 2017.
- [18] S. Krüger and B. Hermann. Can an online service predict gender? On the state-of-the-art in gender identification from texts. In *2019 IEEE/ACM 2nd International Workshop on Gender Equality in Software Engineering (GE)*, pages 13–16. IEEE, 2019.
- [19] T. Kudo and J. Richardson. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In E. Blanco and W. Lu, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018: System Demonstrations, Brussels, Belgium, October 31 - November 4, 2018*, pages 66–71. Association for Computational Linguistics, 2018.
- [20] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *CoRR*, abs/1907.11692, 2019.
- [21] M. Mintz, S. Bills, R. Snow, and D. Jurafsky. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 1003–1011. Association for Computational Linguistics, 2009.
- [22] M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, and M. Auli. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*, 2019.
- [23] J. Read. Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In *Proceedings of the ACL student research workshop*, pages 43–48, 2005.
- [24] A. Siewierska. Gender distinctions in independent personal pronouns. In M. S. Dryer and M. Haspelmath, editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig, 2013.
- [25] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958, Jan. 2014.
- [26] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.
- [27] R. Veenhoven, S. Snijders, D. van der Hall, and R. van Noord. Using translated data to improve deep learning author profiling models. In *Proceedings of the Ninth International Conference of the CLEF Association (CLEF 2018)*, volume 2125, 2018.
- [28] A. Wang, Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. *arXiv preprint 1905.00537*, 2019.
- [29] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. 2019. In the Proceedings of ICLR.

A Comparative Study of Short Text Classification with Spiking Neural Networks

Piotr S. Maciąg, Wojciech Sitek, Łukasz Skonieczny, Henryk Rybiński

Warsaw University of Technology

Institute of Computer Science

Nowowiejska 15/19, 00-665, Warsaw, Poland

piotr.maciag@pw.edu.pl, wojciech.sitek@pw.edu.pl, lukasz.skonieczny@pw.edu.pl, hrb@ii.pw.edu.pl

Abstract—Short text classification is an important task widely used in many applications. However, few works investigated applying Spiking Neural Networks (SNNs) for text classification. To the best of our knowledge, there were no attempts to apply SNNs as classifiers of short texts. In this paper, we offer a comparative study of short text classification using SNNs. To this end, we selected and evaluated three popular implementations of SNNs: evolving Spiking Neural Networks (eSNN), the *NeuCube* implementation of SNNs, as well as the *SNN Torch* implementation that is available as the Python language package. In order to test the selected classifiers, we selected and preprocessed three publicly available datasets: 20-newsgroup dataset as well as imbalanced and balanced PubMed datasets of medical publications. The preprocessed 20-newsgroup dataset consists of first 100 words of each text, while for the classification of PubMed datasets we use only a title of each publication. As a text representation of documents, we applied the TF-IDF encoding. In this work, we also offered a new encoding method for eSNN networks, that can effectively encode values of input features having non-uniform distributions. The designed method works especially effectively with the TF-IDF encoding. The results of our study suggest that SNN networks may provide the classification quality is some cases matching or outperforming other types of classifiers.

Index Terms—short text classification, spiking neural networks, evolving spiking neural networks, *NeuCube*, *SNN Torch*, medical documents, PubMed documents

I. INTRODUCTION

EFFECTIVE text classification is a difficult task that often requires adaptation of special types of learning and encoding methods. Classification of short texts is often even more difficult due to the very limited length of documents that can be used as training input data. The already offered methods for short text classification include, for example, Support Vector Machines (SVM), naive Bayes classifier, decision trees [1] or the classifiers that include clustering of input data as a preprocessing step [2].

Spiking Neural Networks (SNNs) are a type of neural networks that are highly inspired by biological mechanisms of learning and cognition of a human brain. Surprisingly, there are no publications applying SNNs to short text classification. Thus, in this work we evaluate three selected implementations of SNNs applied by us to the short text classification task, namely: our prepared classifier that uses *evolving Spiking Neural Networks*, *eSNNs*, the *NeuCube* implementation of SNNs as well as the *SNN Torch* implementation.

Evolving Spiking Neural Network (eSNN) is a recently introduced classifier that was successfully applied in various domains: transportation prediction [3], air pollution prediction [4]–[6], recognition of moving objects [7], or anomaly detection [8], [9]. To the characteristic of eSNNs belongs: ability to process large amounts of data efficiently and insignificant memory requirements. As it was proven in the enumerated examples of eSNNs usage cases, they can effectively use the biologically inspired learning and prediction mechanisms in typical engineering applications.

The other implementation selected for this study is the *NeuCube* implementation of SNNs [10], [11]. Contrary to the eSNNs implementation, *NeuCube* consists of three layers of spiking neurons: input, whose aim is to encode input values into firing times, internal, which consists of a reservoir (cube) of hidden neurons, whose weights are trained in an unsupervised manner using synaptic-plasticity rules, and output, which contains neurons responsible for assigning decision classes to testing examples.

Finally, as the third implementation of SNNs we selected recently developed *SNN Torch* implementation available as a package of the Python language [12]. To the advantages of *SNN Torch* belong its flexibility to construct SNNs that can consist of many layers of neurons which combine not only neuronal models that are typically present in SNNs, such as the Leaky-Integrate-and-Fire (LIF) model, but also sigmoid neuronal models. In addition, *SNN Torch* can take advantage of a GPU-based processing in order to speed up training and classification procedures.

This paper provides the following contribution:

- To the best of our knowledge, for the first time in the literature we apply SNNs for classification of short texts and, especially, large sets of medical publications based on their metadata (such as a title or an abstract of a publication). To this end, we selected three types of SNNs: *eSNN* networks, the *NeuCube* implementation of SNNs as well as the *SNN Torch* implementation.
- As a part of our implementation of *eSNN* networks we propose a new input data encoding method. The proposed method first creates a histogram of input values of each feature F in the training dataset. The number of bins (subranges) of histogram is specified by a user-given parameter called B . Subsequently, the NI_{size} input

neurons of an eSNN are redistributed to encode the values of each bin of the histogram according to the cardinality of values in bins. As we present in the experiments, the offered encoding method provides much better classification accuracy than the other two encoding methods offered in the literature: a method that directly calculates the firing order of input neurons proposed in [5] and Gaussian Receptive Fields (GRFs) [13].

- We conduct experiments using the frequently-used *20-newsgroup* dataset¹ as well as two real PubMed datasets of medical publications selected from the website of the BioASQ competition². Since we focus on classifying short texts, from each document of the *20-newsgroup* we selected only 100 first words. In the case of two selected PubMed datasets, only a title of a publication is used as input data for each tested classifier.
- The obtained results of experiments suggest that SNN Torch implementation is more effective in short text classification than the other selected SNNs implementations. Additionally, for the selected PubMed datasets, SNN Torch gives results of classification slightly superior to the other classifiers tested in the experiments.

The paper is structured as follows. Section II presents the related work. Section III describes the SNNs implementations selected for this study. This section also describes the proposed encoding method for the eSNN networks. In Section IV, we give the description of the obtained datasets and applied preprocessing. Section V provides the results of experiments. Finally, in Section VI we conclude the work and discuss the results.

II. RELATED WORK

A. Short Text Classification

Effective classification of short texts is a topic intensively studied nowadays. The already offered methods offered for text classification include: various types of classifiers, such as Support Vector Machines (SVM), naive Bayes classifiers, decision trees or different types of neural networks [1]. [14] distinguishes two types of approaches that can be applied for text classification: the first one, in which the set of text is first represented using Document-Term Matrix (DTM), which can be obtained using feature extraction method, such as Bag of Words (BoW) or Term Frequency - Inverse Document Frequency (TF-IDF). Subsequently, DTM is used to train a selected classifier, such as SVM. The second approach skips the process of generating DTM matrix and directly provides the set of texts as training data for a deep neural network.

Majority of recent approaches to short text classification with neural network models focused on applying Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs). [15] presented a classification model that is based on CNNs and incremental learning. In order to increase the classification accuracy, [15] applied an approach in which a

current short document of a pipeline of documents is classified not only based on its textual content but also based on obtained classification results of preceding documents.

[16] presented the results of experiments on the classification of one-sentence questions using CNNs. The dataset applied in [16] was obtained from the WikiAnswer and contained 608 650 questions grouped into several hundred categories. As the results of experiments of [16] suggest, CNN networks can classify questions with accuracy comparable or better than SVM classifier.

In [17], a model that combines CNNs networks with SVM classifier was applied to the classification of simple sentences expressing either positive and negative feelings. In the approach presented in [17], first, a word embedding method (such as Word2Vec) is used to obtain the vector representation of each word in the text corpus. Subsequently, each text in the corpus is represented as a sequence of vectors, each corresponding to one of the text's words. Such a representation of texts is next used as input data for CNN network that consists of convolutional, max-pooling and fully-connected layers. The results obtained from the fully-connected layer are used as training data for the SVM classifier. The results of experiments presented in [17] suggest that this approach can provide better classification results than separate CNNs and SVM classifiers.

The review of the other types of classifiers applied for (short) text classification (and in particular MESH dataset) can be found, for example, in [14], [18], [19].

B. Spiking Neural Networks for Text Classification

We are aware of only two other works adapting spiking neural networks for text classification. However, contrary to this work, both of them applied SNNs for long text classification. In order to classify longer texts, [20] offered a method consisting of two phases. The first phase consists of transforming a text into a vector of numbers using TF-IDF encoding and, subsequently, into a sequence of spikes. In the second phase, an SNN network is taught in an unsupervised way using the spikes generated in the first phase in order to generate spike-based low-dimensional representation of a text. Subsequently, the generated representation is used as input data for the training of logistic regression, which is responsible for the final classification of documents. Thus, in the approach of [20], an SNN network can be perceived as a dimensionality reduction technique of a TF-IDF text representation. While the approach of [20] was shown to provide superior text classification results to the other classifiers tested there, it used only logistic regression, which (as presented, for example, in our experiments) itself can be an effective text classifier.

[21] presents a comparison of classification results for different types of word embeddings (such as Word2Vec and GloVe [22]) and neuron types that were applied in SNN (such as the LIF neuronal model). The results obtained in [20] suggest that SNNs can be effectively applied to classify longer texts.

¹<http://qwone.com/~jason/20Newsgroups>

²<http://participants-area.bioasq.org/datasets>

III. THE SELECTED IMPLEMENTATIONS OF SPIKING NEURAL NETWORKS

A. The Evolving Spiking Neural Networks Implementation

In Fig. 1, we present the architecture of our implementation of eSNNs. The designed eSNN network consists of groups of input neurons $\text{NI}^{(F_1)}, \dots, \text{NI}^{(F_m)}$ encoding values of m features $\mathcal{F} = \{F_1, \dots, F_m\}$ ³. The number of input neurons in each group $\text{NI}^{(F)}$ is the same and is specified by the user-given parameter NI_{size} . The output neurons in the repository NO are assigned decision classes present in the training dataset of texts \mathbf{D}_{tr} (we assume that each text in $T \in \mathbf{D}_{tr}$ has one and only one decision class). Thus, given L decision classes, the output neurons NO are organized into L groups. In the learning process of an eSNN network, a new candidate output neuron n_c is created for each training text $T \in \mathbf{D}_{tr}$ and either added to the repository NO or merged with the neurons already existing there.

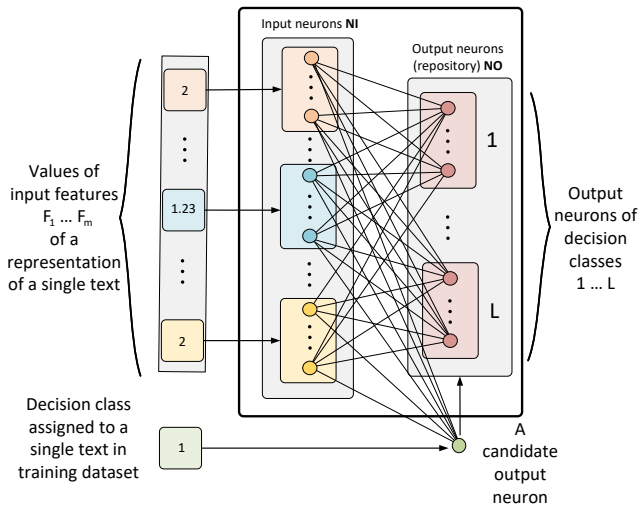


Fig. 1. The architecture of an eSNN network adopted in this study.

1) *Input Layer*: For the encoding of input values of features \mathcal{F} into spikes we develop a new encoding method. Our motivation to develop the presented method lies in the fact that the previously introduced methods dedicated for eSNNs⁴ (such as the GRFs method [24] or the method of [5]) do not work well with some text representations (such as the TF-IDF representation)⁵. Specifically, both the GRF method and the method of [5] are not able to effectively encode input values of features having non-uniform distributions. This can

³The number of features \mathcal{F} depends on the used text representation: for example, Word2Vec and Doc2Vec representations allow the user to specify the number of features [23].

⁴Unlike the other implementations of SNN networks, eSNN does not require exact firing times of input neurons to be propagated into the network. Rather than, it is enough to calculate the firing order of input neurons $\text{NI}^{(F)}$ encoding a value of each feature $F \in \mathcal{F}$.

⁵In fact, in Section V we present experiments comparing the mentioned encoding methods for eSNNs.

be explained by the fact that both of these methods divide the range of input values of a feature F into a number of equal subranges, the number of which is equal to the user-given number of input neurons NI_{size} . A center value of each of such subranges is associated with one input neuron. Given an input value to be encoded and the obtained center values of input neurons, the GRFs method and the method of [5] calculate Euclidean distances between the input value and the center values of input neurons. The non-decreasing order of such Euclidean distances between the input value and the center values of input neurons identifies the firing order of input neurons. In the case of non-uniform distributions of input values (such as a normal distribution or skewed distributions), it can often happen that a relatively high number of input values will be encoded using the same firing order of input neurons, while the values of other ranges will be associated with more distinguishing firing order of input neurons. To alleviate this problem, we offer the method presented below.

The proposed encoding method requires two user-given input parameters, which are the number of input neurons NI_{size} encoding values of each feature $F \in \mathcal{F}$ as well as the number of bins (subranges) B (where $B < NI_{size}$) used to create a histogram of values of each feature F . The proposed method first creates a histogram of input values of a feature F using solely the training dataset. Subsequently, the NI_{size} input neurons are allocated to encode the values of each bin of the histogram in the following way. First, each bin is assigned at least one input neuron of NI_{size} neurons. The rest of $NI_{size} - B$ input neurons is allocated as follows.

Let $Min^{(F)}$ and $Max^{(F)}$ be minimal and maximal values of feature F in training dataset \mathbf{D}_{tr} , respectively. The width of range of each bin equals $Bins_{width} = \frac{Max^{(F)} - Min^{(F)}}{B}$. The range of each bin is calculated as follows:

- $[Min^{(F)} + (i - 1) \cdot Bins_{width}, Min^{(F)} + i \cdot Bins_{width})$, for $Bin_i, i = \{1, \dots, B - 1\}$.
- $[Min^{(F)} + (B - 1) \cdot Bins_{width}]$, for Bin_B .

Subsequently, the number of values of a feature F in each bin is calculated and remembered as $Bin_i.\overline{Values}$. The number of neurons allocated to each bin is obtained using Proposition 1.

Proposition 1. Let $Bin_i.\overline{Neurons}$ represents a number of input neurons allocated to encode values of Bin_i and $\overline{\mathbf{D}_{tr}}$ be a number of training examples (texts):

- $Bin_i.\overline{Neurons} = \left\lceil \frac{Bin_i.\overline{Values}}{\overline{\mathbf{D}_{tr}}} \cdot (NI_{size} - B) \right\rceil + 1$, for $i \in \{1, \dots, B - 1\}$,
- $Bin_i.\overline{Neurons} = \left\lceil \frac{Bin_i.\overline{Values}}{\overline{\mathbf{D}_{tr}}} \cdot (NI_{size} - B) \right\rceil + 1$, for $i = B$.

Given the number of input neurons allocated to each bin, we can obtain the center value μ_j of each input neuron $n_j \in \text{NI}$. The center values μ_j of input neurons are directly used to calculate firing order of input neurons. The center values are calculated according to Proposition 2. For each Bin_i , let

$$Bin_i.\Delta = Bins_{width} / Bin_i.\overline{Neurons}$$

denote the width between center values of input neurons allocated to Bin_i .

Proposition 2. For each bin $i \in \{1, \dots, B\}$, the center value of each input neuron $n_j \in Bin_i.Neurons, j \in \{1, \dots, Bin_i.Neurons\}$ is calculated as follows

$$\mu_j = Min^{(F)} + (i - 1) \cdot Bins_{width} + (j - 0.5) \cdot Bin_i \cdot \Delta$$

Finally, given the center values of input neurons (please note, that it is enough to calculate the center values μ_j one time after D_{tr} is load by an eSNN), we can calculate firing order $order_{n_j}$ of spikes that are propagated into the network. Let assume that $x^{(F)}$ is the value of a feature F to be encoded by the proposed method. Proposition 3, shows how to obtain the center value closest to an input value $x^{(F)}$.

Proposition 3. Let μ_k be the center value closest to input value $x^{(F)}$. Index k is calculated as follows:

$$k = \begin{cases} j \mid |x^{(F)} - \mu_j| \text{ is smallest,} & \text{if } x^{(F)} \in [Min^{(F)}, Max^{(F)}] \\ 1, & \text{if } x^{(F)} < Min^{(F)}, \\ NI_{size}, & \text{if } x^{(F)} > Max^{(F)}. \end{cases}$$

Given the k index, the firing order of all input neurons $n_j \in NI$ is obtained as given in Algorithm 1. We illustrated the example coding using the proposed method in Fig. 2.

Algorithm 1 Calculate firing order of input neurons

Input: $x^{(F)}$ - input value to be encoded, NI_{size} - number of input neurons.

Ensure precalculated: Bin_i - the structure containing parameters of i -th bin of a feature F (i.e. $Bin_i.Neurons, Bin_i.\Delta, Bin_i.Values$), μ_j - center values of input neurons $n_j \in \{1, \dots, NI_{size}\}$, k - index of a center value μ closest to $x^{(F)}$.

Output: Firing order of input neurons $n_j \in \{1, \dots, NI_{size}\}$.

```

1:  $order_{n_k} \leftarrow 0, ord \leftarrow 0$ 
2:  $l \leftarrow k - 1; r \leftarrow k + 1$ .
3: while  $l \geq 1$  OR  $r \leq NI_{size}$  do
4:   if  $l \geq 1$  then  $dist_l \leftarrow |\mu_l - x^{(F)}|$  end if
5:   if  $r \leq NI_{size}$  then  $dist_r \leftarrow |\mu_r - x^{(F)}|$  end if
6:   if  $l < 1$  AND  $r \leq NI_{size}$  then
7:      $order_{n_r} \leftarrow ord, ord \leftarrow ord + 1, r \leftarrow r + 1$ 
8:   else if  $l \geq 1$  AND  $r > NI_{size}$  then
9:      $order_{n_l} \leftarrow ord, ord \leftarrow ord + 1, l \leftarrow l - 1$ 
10:  else if  $l \geq 1$  AND  $r \leq NI_{size}$  then
11:    if  $dist_l < dist_r$  then
12:       $order_{n_l} \leftarrow ord, ord \leftarrow ord + 1, l \leftarrow l - 1$ 
13:    else
14:       $order_{n_r} \leftarrow ord, ord \leftarrow ord + 1, r \leftarrow r + 1$ 
15:    end if
16:  end if
17: end while

```

Algorithm 1 calculates firing order of input neurons as follows. As a first fires the input neuron n_k , whose center value μ_k is closest to the input value $x^{(F)}$ (the firing order function $order_{n_k}$ of the input neuron n_k is set to 0). Subsequently,

Algorithm 1 calculates firing order of the rest input neurons, whose center values are located to the left and to the right of the center value of the first firing input neuron n_k . The firing order of these neurons is calculated in a single scan using the distances between their center values and the input value $x^{(F)}$. To this end, the algorithm uses three counters: l and r which point to the neurons whose center values are located to the left and to the right of the center value μ_k , respectively, as well as ord counter which stores the current firing counter (initially set to 0). Initially, l and r are set to $k - 1$ and $k + 1$, respectively. In each iteration of the main while loop, the algorithm calculates firing order of one input neuron pointed either by l or r counters. If the distance between center value of an input neuron n_l and $x^{(F)}$ is smaller than the distance between center value of an input neuron n_r and $x^{(F)}$, then $order_{n_l}$ is set to the current value of the ord counter, ord is incremented and l is decremented. Otherwise, $order_{n_r}$ is set to ord , ord is incremented and r is incremented. The computational complexity of Algorithm 1 is linear in the number of input neurons NI_{size} , similarly to the encoding algorithms presented in [5] or [13].

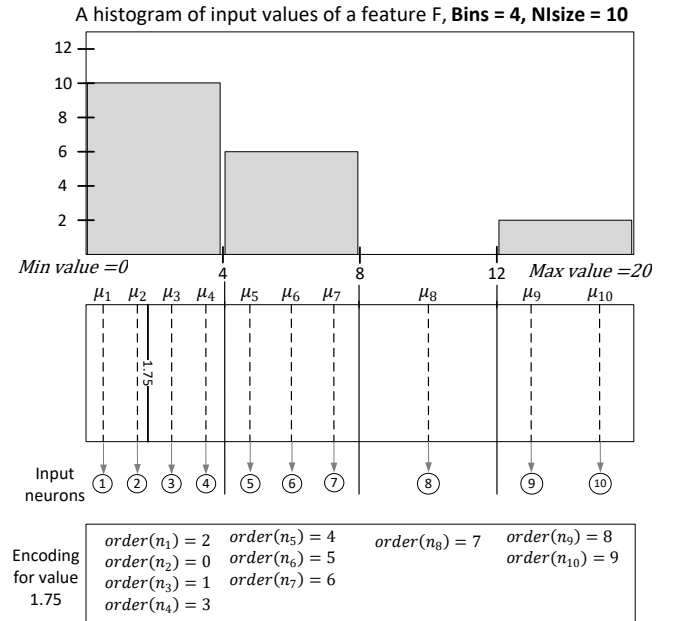


Fig. 2. The proposed encoding method - the NI_{size} input neurons are redistributed to code the values of each bin of a histogram; the encoded value is 1.75.

2) *Network's Learning and Classification:* The firing orders of input neurons are calculated separately for each text in either training or testing dataset. The firing orders obtained for texts of the training dataset are used in the network's learning phase, while the firing orders calculated for texts of the testing dataset are used to classify these texts.

In the eSNN learning process, for each training text T in dataset D_{tr} , there is created a candidate output neuron n_c

which is assigned a single decision class of text T . We denote a decision class assigned to n_c as: $Class(n_c)$.

The candidate output neuron is connected through synapses to all input neurons \mathbf{NI} . The vector of weights of such synapses is denoted as $\mathbf{w}_{n_c} = [w_{n_1, n_c}^{(A_1)}, \dots, w_{n_{|\mathbf{NI}(F_1)|}, n_c}^{(F_1)}, \dots, w_{n_1, n_c}^{(F_m)}, \dots, w_{n_{|\mathbf{NI}(F_m)|}, n_c}^{(F_m)}]$, where m is the number of features \mathcal{F} . To initialize the weights of synapses we apply the rank-order rule [24]. Each weight of a vector \mathbf{w}_{n_c} is initialized according to Eq. (1).

$$w_{n_j n_c}^{(A)} = mod^{order_{n_j}}, n_j \in \mathbf{NI}, \quad (1)$$

where mod is a modulation factor whose value is specified by the user and should be in the range $(0, 1)$.

Each output neuron (either candidate n_c or the output neuron n_i already present in \mathbf{NO}) has also an update counter M . The value of such update counter is first set to 1 when a candidate is created, and subsequently is incremented when an output neuron present in \mathbf{NO} is updated using a candidate output neuron.

After the candidate neuron n_c is created and its synapses' weights are initialized, it is either added to the repository of output neurons \mathbf{NO} or merged with one of the output neurons already existing in $\mathbf{NO}(Class(n_c))$, that is in the group of output neurons of class $Class(n_c)$ in \mathbf{NO} . To this end, Euclidean distances $Dist_{n_c, n_i}$ between the vector of synapses' weights \mathbf{w}_{n_c} and the vectors of synapses weights \mathbf{w}_{n_i} of each output neuron $n_i \in \mathbf{NO}(class_{n_c})$ are calculated. If there exists such an output neuron n_s for which $Dist_{n_c, n_s}$ is minimal and below the value $simTr \cdot \overline{Dist}$, then the vector \mathbf{w}_{n_s} and counter M_{n_s} are updated according to Eq. (2) and n_c is discarded. Otherwise, n_c is simply inserted into $\mathbf{NO}(class_{n_c})$. $simTr$ is a user-specified similarity threshold, whose value is in the range $[0, 1]$ ⁶.

$$\mathbf{w}_{n_s} = \frac{\mathbf{w}_{n_s} \cdot M_{n_s} + \mathbf{w}_{n_c}}{M_{n_s} + 1}, M_{n_s} = M_{n_s} + 1. \quad (2)$$

\overline{Dist} is the tight upper bound on the Euclidean distances between any possible candidate output neuron and any output neuron in \mathbf{NO} and, as presented in [5], can be calculated using Eq. (3).

$$\overline{Dist} = \left[\sum_{F \in \mathcal{F}} \sum_{j=1}^{NI_{size}} \left(mod^{j-1} - mod^{NI_{size}-j-1} \right)^2 \right]^{\frac{1}{2}} \quad (3)$$

After eSNN is taught using the training dataset \mathbf{D}_{tr} , each testing text $T \in \mathbf{D}_{ts}$ is assigned one decision classes of all decision classes of output neurons in \mathbf{NO} . To this end, the value of Post-Synaptic Potential PSP_{n_i} of a membrane of each output neuron n_i in \mathbf{NO} is calculated according to Eq. (4).

$$PSP_{n_i} = \sum_{F \in \mathcal{F}} \sum_{n_j \in \mathbf{NI}^{(F)}} w_{n_j n_i}^{(F)} \cdot mod^{order_{n_j}}. \quad (4)$$

⁶Please note, that the greater values of $simTr$ increase the chance that a candidate output neuron will be merged with one of the neurons already existing in the repository \mathbf{NO}

In Eq. (4), $w_{n_j n_i}$ is weight of a synapse connecting the input neuron n_j to the output neuron n_i that is calculated in the network's learning phase. $order_{n_j}$ is a firing order value of input neuron $n_j \in \mathbf{NI}^{(A)}$ given the encoding of a value of feature F in a testing text T . Finally, the testing text T is assigned a decision class of an output neuron $n_{max} \in \mathbf{NO}$, whose membrane PSP value $PSP_{n_{max}}$ is maximal. One can find the pseudocode of described learning and classification procedures of eSNNs, for example, in [5], [9], [24]. We posted our implementation that was used in the experiments along with the used dataset at the GitHub repository⁷.

B. The NeuCube Implementation of Spiking Neural Networks

As the second implementation of SNNs that is selected by us for the experiments we used the NeuCube implementation [10]⁸. Unlike the eSNNs implementation presented in subsection III-A, NeuCube implements an SNN network that consists of three layers of neurons: input, internal and output. The aim of the input layer of NeuCube is to convert values of features of a text representation into a sequence of spikes that is propagated into the network.

Since NeuCube implements four temporal coding algorithms, it requires each text (either from the training \mathbf{D}_{tr} or testing \mathbf{D}_{ts} datasets) to be represented as a time series. In our approach, each text is represented as a single time series TS containing all values of features (F_1, \dots, F_m) . Thus, given a text representation having m features \mathcal{F} , the time series of each text consists of a series of m values. The temporal encoding algorithms implemented in NeuCube are: Threshold-based Representation (TR), Moving Window (MV), Step Forward (SF), and Bens Spiker Algorithm. The results presented in [25] suggests that the most effective is the TR algorithm, which was used by us in the experiments. Given time series representation TS of a text, the TR algorithm generates spikes by first calculating $ATB = \mu SR \cdot \sigma$ value (where μ and σ are mean and standard deviation of values of TS , respectively). Next, a positive spike is generated if the difference between two consecutive values of TS is positive and greater than ATB . If the difference is negative and smaller than ATB , then a negative spike is generated. The generated example of TR encoding of the time series values of the first document of the 20-newsgroup dataset is given in Fig. 3.

The internal layer of NeuCube consists of a cube of Leaky-Integrate-and-Fire (LIF) neurons [26]–[28] that are interconnected using both excitatory and inhibitory synapses. The number of such neurons in the cube and their topological locations can be defined by the user. The initial synapses and their weights are generated using the *small-world* principle (according to which the neurons located in a topological proximity have a grater chance to be connected). In the cube's learning process, the weights of synapses are calculated according to the Spike-Time Dependent Plasticity (STDP)

⁷<https://github.com/piotrMaciag32/eSNN-short-text-classifier>

⁸NeuCube is a an application with a graphical user interface implemented in Matlab and is free for download form <https://kedri.aut.ac.nz/R-and-D-Systems/neucube>

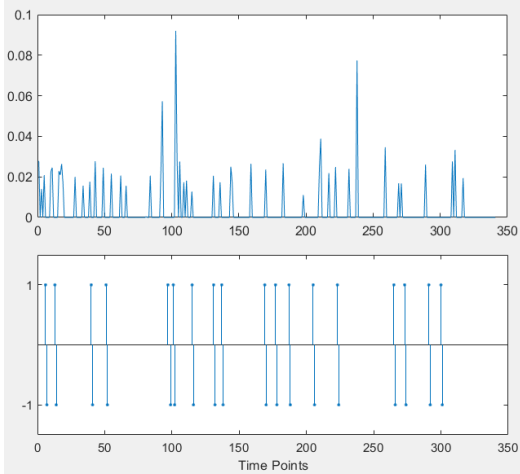


Fig. 3. The NeuCube TR encoding of the time series values of the first document of the 20-newsgroup dataset (the upper plot presents an input time series of TF-IDF values, the lower plot shows the obtained encoding).

rules. Let us consider two neurons: n_j and n_i presented in the internal layer of NeuCube and let us assume that there is a synapse from neuron n_j to neuron n_i . Given emission of spikes from neurons n_j and n_i at times t_j and t_i , respectively, the change of the weight value of the synapse from n_j to n_i is calculated according to Eq. (5).

$$w_{j,i}(t) = \begin{cases} w_{j,i}(t-1) + \eta/\Delta t, & \text{if } t_i > t_j, \\ w_{j,i}(t-1), & \text{if } t_i = t_j, \\ w_{j,i}(t-1) - \eta/\Delta t, & \text{otherwise,} \end{cases} \quad (5)$$

where η is the STDP rate learning parameter specified by the user.

Finally, the third layer of NeuCube consists of output neurons whose aim is to represent decision classes present in the training dataset \mathbf{D}_{tr} . Each output neuron in the output layer of NeuCube is connected to all neurons in the internal layer. The output neurons in NeuCube are grouped according to decision classes similarly to the output neurons **NO** of eSNNs. As in eSNNs, for each training text there is created one candidate output neuron that is always added to the set of output neurons of NeuCube (unlike in eSNNs, in which candidate output neurons can be merged with the output neurons already existing in the output layer).

The membrane Post-Synaptic Potentials (PSP) values of both internal and output neurons are calculated according to Eq. (4). Both internal and output neurons emit a spike when their PSP values exceed a certain firing threshold C , which is specified by the user. Specifically, a neuron n_i emits a spike according to Eq. 6.

$$\text{Emit a spike by } n_i \text{ at time } t = \begin{cases} True & \text{if } PSP_{n_i} \geq C, \\ False & \text{if } PSP_{n_i} < C, \end{cases} \quad (6)$$

In Fig. 4, we present the architecture of NeuCube along with the selected representation of each text. In Table I, we

show the learning parameters of NeuCube along with their description.

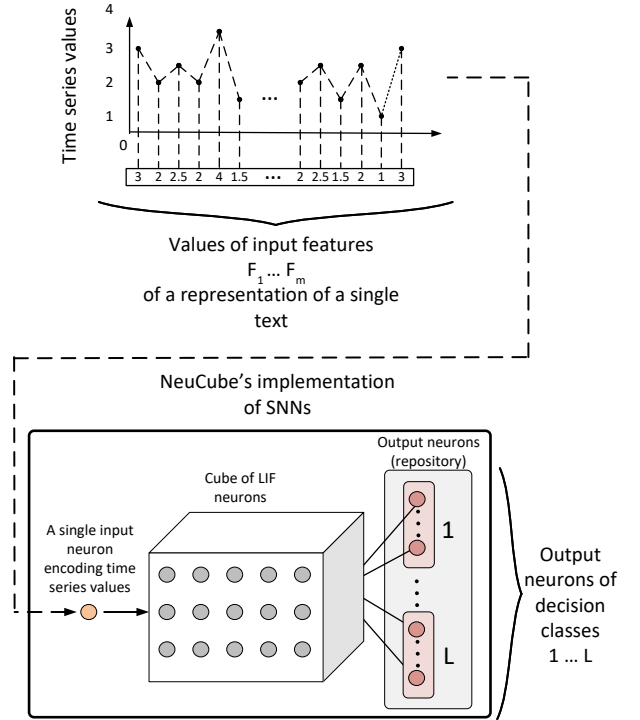


Fig. 4. The NeuCube's architecture and the applied representation of a text as a series of TF-IDF values.

C. The SNN Torch Implementation

The third implementation of SNNs selected for the experiments is the SNN Torch implementation, recently developed as the Python language package [12]. To the advantages of the SNN Torch implementation belongs the fact that it is built on the basis of the well known deep-learning Python framework *Pytorch*. SNN Torch allows us to combine SNNs with such types of neural networks as Multilayer Perceptron neural network or CNN network. Currently, SNN Torch offers eight types of spiking models of neurons: Alpha, Lapique, Leaky (LIF), RLeaky, RSynaptic, SConv2dLSTM, SLSTM and Synaptic. In our experiments, we applied the LIF neuronal model. Our applied architecture of neurons in SNN Torch consists of four layers as presented in Fig. 5. The number of input neurons equals the number of features in the input data, while the number of output neurons is the same as the number of decision classes present in the training part of the dataset.

IV. CHARACTERISTIC OF SELECTED DATASETS AND THEIR PREPROCESSING

In the experiments, we used three publicly available datasets that are widely used as benchmarks in the evaluation of text classifiers. In the experiments, we applied the TF-IDF representation of all texts. As previous experiments with text

TABLE I
THE MAIN PARAMETERS OF SNN USED IN NEUCUBE.

Parameter	Description
SR	TR algorithm threshold.
η (STDP Rate)	STDP rate for weights modification in the internal layer.
Refractory time	A period in which an input is inactive to incoming spikes after emission of a spike by itself.
Mod	Modulation parameter as given in Eq. 5.
Training iters.	Number of its. of the unsupervised learning stage.
Firing treshold	Firing threshold for spike emission by neurons.

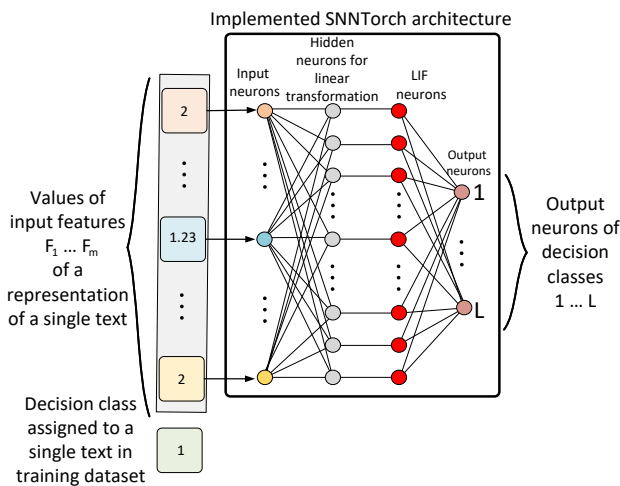


Fig. 5. The SNN architecture that was constructed using SNN Torch package for the purpose of experiments.

data and SNN networks suggest (see, for example, [20]), TF-IDF is a suitable representation for these type of neural networks. In order to obtain the TF-IDF representation, first, for each dataset we calculated the Document-Term Matrices (DTMs). DTM is a matrix that contains as many rows as the number of documents in either training or testing part of the dataset, and as many columns as the size of vocabulary (the set of all terms present in the entire dataset). Each cell of a DTM matrix can contain, for example, a number of occurrences of a given word of vocabulary (defined by a column of DTM) in a document (defined by a row of DTM)⁹.

For all selected datasets, the DTM matrices are obtained as follows. First, the vocabulary is calculated. In order to obtain the vocabulary of each dataset, we applied the *text-mining* package of the R language [29] (for the 20-newsgroup dataset) and *Gensim* package of the Python language [30] (for the PubMed Mesh dataset). The vocabulary is obtained by applying the following steps to the set of texts of each dataset:

- 1) Removing all numbers from a text.
- 2) Removing punctuation.

⁹Obviously, such a representation usually leads to a significant size of a DTM, whose cells mostly contain 0 (which indicates the situation when a word is not present in a document).

- 3) Removing English stopwords and articles.
- 4) Transforming all upper-case letters to lower-case letters.
- 5) Applying texts' stemming.

After the execution of the above steps, we calculated the DTM matrices for the training and testing parts of datasets separately as follows.

A. The Preprocessed 20-newsgroup Dataset

The 20-newsgroup dataset contains 18 846 texts that are grouped into 20 news categories. To the distinguishing characteristic of the 20-newsgroup dataset belongs the fact that 20 categories often belong to very different domains. For example: politics, sociology, religion or computer devices. The texts are split into training and testing parts in the proportion 6:4 by the author of the dataset. Thus, the training part consists of 11 307 training texts and 7538 testing texts. Most of the texts in the datasets consists of several hundreds of words. Thus, we have shorten each text by selecting only its 100 first words as text used for classification. The vocabulary calculated using such preprocessed shorten texts contains 132 370 terms. The short texts are used to obtain the Document Term Matrices (DTMs) for training and testing parts separately. Since the obtained vocabulary contains 132 370 terms, each DTM matrix would also contain 132 370 columns - far too many for most of classifiers. Thus, we decided to remove sparse words from DTMs as follows:

- For the NeuCube implementation we remove from the vocabulary these terms which are not present in at least 95% of text (this reduces the number of columns of DTM to 341) - such a reduction was forced by the memory constraint of NeuCube, which prevents loading too large datasets.
- For all other tested classifiers we remove from the vocabulary these terms which are not present in at least 99% of text (this reduces the number of columns of DTM to 751).

B. The Preprocessed Imbalanced and Balanced PubMed Dataset

The PubMed dataset contains several millions of medical publications that are categorized according to the Medical Subject Headings (MeSH). MeSH is a set of classes organized into a hierarchical structure with 16 main branches (in overall, MeSH consists of nearly 30 000 categories). Each PubMed

document is usually indexed (either by the authors of a document/publication or by a publisher) using several MeSH categories. The obtained by us metadata of PubMed documents is posted as a part of the BioASQ competition [31].

For the purpose of our experiments, we randomly selected metadata of 10 000 PubMed documents, each of which is assigned one of the 16 main categories of the MeSH classification. Since the documents in these main categories are unevenly distributed, the resultant dataset is *imbalanced* (for example, majority of publications belong to the category *Chemicals and Drugs*, while there are few publications belonging to the category *Information Sciences*). Since we are focused on classification of short texts, we decided that input data provided to classifiers will contain only a title of a publication. The selected 10 000 publications are split into a training and testing parts in the ratio 9:1.

We applied the TF-IDF encoding to obtain the DTM matrices of the training and testing parts according to the steps given at the beginning of this section. The vocabulary consists of 14553 terms from which we selected 1670 terms that occur in at least 99.9% of texts to represent input features of datasets.

In a similar way, we obtained the *balanced* PubMed dataset, which differs from the imbalanced in that it contains the 10 000 documents that are evenly distributed in the 16 main categories.

V. EXPERIMENTS

In this section, we first describe the applied input parameters of the selected classifiers. Next, we present the results of experiments on the selected datasets. The results were obtained for the three above-described implementations of SNNs as well as for the other four classification methods: Binomial logistic regression, a single Decision tree, the MLP neural network as well as Support Vector Classifier (SVC). In the second part of the experiments, we focus specifically on the presentation of the classification accuracy for the eSNN encoding method that is offered by us in Section III.

A. Parameters of Selected Classifiers

The parameters of the selected SNNs implementations that were ran on the datasets are given in Table II. The parameters were selected using the grid search procedure on a suitable set of parameters of each implementation.

The parameters of the other classifiers selected for experiments were as follows:

- Decision tree - split criterion = Gini index, maximal depth = none.
- Logistic regression - maximal no. of training iterations = 100.
- Multilayer Perceptron neural network (MLP) - no. of hidden neurons = 1000, sigmoid activation function, learning rate for weights modification in thw error backpropagation phase 0.0001, training iterations = 8, the ADAM optimization method for error backpropagation.
- Support Vector Classifier - radial kernel, misclassification cost coefficient = 1.

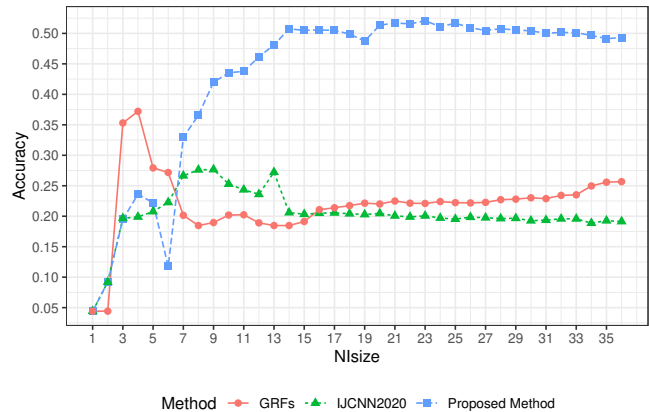


Fig. 6. Comparison of the classification accuracy for the selected input methods. *IJCNN2020* refers to the method offered in [5]. The method given in this work was ran with $Bins = 3$.

B. Results of Short Text Classification

The results of classification accuracy for the classifiers are given in Table III. As it can be noted, in the case of short texts of the 20-newsgroup dataset, the most effective classifier is SVC. For both balanced and imbalanced PubMed datasets, two best performing classifiers are SNN Torch and logistic regression. Among the selected SNNs implementations, the eSNN implementation provides slightly better results than NeuCube for 20-newsgroup dataset, while NeuCube is slightly superior to eSNNs in the cases of PubMed datasets. The best performing implementation of SNNs is SNN Torch. This can be explained by the fact that it contains not only spiking LIF neurons, but also incorporates learning mechanisms of traditional feedforward neural networks, such as the error backpropagation phase that applies ADAM optimizer.

C. Comparison of the Classification Accuracy for the Proposed Encoding Method of eSNNs

Our method is compared with the encoding method offered in [5] that directly calculates firing order of input neurons as well as with the widely-used GRFs method used with spiking neural networks [13]. Both the method of [5] as well as the GRFs uniformly allocate the NI_{size} input neurons to encode input values of each feature $F \in \mathcal{F}$. In this experiments we use 20-newsgroup dataset, however to better illustrate the results of encoding we selected full texts of this dataset.

In Fig. 6, we present the obtained classification accuracy of the selected methods for varying number of input neurons NI_{size} . For the proposed method, we used the $B = 3$ bins parameter to create the histogram of values of each feature in our encoding method. As it can be noticed from the figure, the offered method can significantly improve the classification accuracy comparing to the other two tested methods. For example, for $NI_{size} = 20$ the proposed method gives accuracy 0.52, while the method of [5] and GRFs method give 0.2 and 0.21 accuracy values, respectively.

TABLE II
THE APPLIED PARAMETERS OF THE SELECTED SNN CLASSIFIERS.

Classifier	20 newsgroup dataset	Imbalanced PubMed dataset	Balanced PubMed dataset
eSNN	$N_{Isize} = 25, Bins = 3,$ $mod = 0.95, simTr = 0.05,$	$N_{Isize} = 15, Bins = 3,$ $mod = 0.95, simTr = 0.05$	$N_{Isize} = 15, Bins = 3,$ $mod = 0.95, simTr = 0.05$
NeuCube	$\eta = 0.01, Firing\ thr. = 0.5,$ $mod = 0.95, Refractory\ time = 3,$ Training its. = 1	$\eta = 0.01, Firing\ thr. = 0.5,$ $mod = 0.4, Refractory\ time = 8,$ Training its. = 1	$\eta = 0.01, Firing\ thr. = 0.5,$ $mod = 0.4, Refractory\ time = 8,$ Training its. = 1
SNNTorch	Hidden neu. = 1000, Firing thr. = 1, Decay rate = 0.92, Learn. rate = 0.0001, Training its. = 8	Hidden neu. = 1000, Firing thr. = 1, Decay rate = 0.92, Learn. rate = 0.0001, Training its. = 8	Hidden neu. = 1000, Firing thr. = 1, Decay rate = 0.92, Learn. rate = 0.0001, Training its. = 8

TABLE III
THE OBTAINED CLASSIFICATION ACCURACY RESULTS.

Classifier	20 newsgroup	Imbalanced PubMed	Balanced PubMed
eSNN	0.19	0.15	0.16
NeuCube	0.10	0.17	0.29
SNNTorch	0.24	0.39	0.39
Decision tree	0.43	0.25	0.26
Logistic regression	0.55	0.36	0.32
MLP network	0.01	0.32	0.31
SVC	0.61	0.37	0.37

VI. DISCUSSION AND CONCLUSIONS

In this work, we presented the results of short text classification using three different implementations of SNNs networks, namely: evolving Spiking Neural Networks, the NeuCube implementation of SNNs and the SNNTorch implementation. In order to test the selected classifiers, we selected and preprocessed three publicly available datasets: 20-newsgroup dataset as well as imbalanced and balanced PubMed datasets of medical publications. The preprocessed 20-newsgroup dataset consists of the first 100 words of each text, while for the classification of PubMed datasets we used only a title of each publication. As a text representation of documents, we applied the TF-IDF encoding. In this work, we also offered a new encoding method for eSNN networks, that can effectively encode unevenly distributed values of each input feature. The designed method works especially effectively with the TF-IDF encoding.

The presented results of experiments indicate, that SNNs implementations that solely use the neuronal models tradi-

tionally applied in SNNs, such as the LIF model, as well as apply unsupervised learning rules like STDP, may not perform as effectively as the implementations that combine SNNs with the learning methods present in traditional neural networks, such as the MLP networks. Specifically, in the conducted experiments, the SNNTorch implementation performed better than the eSNN and NeuCube implementations. Furthermore, the computational and memory complexity of SNN networks (as in the case of NeuCube) can be a bottleneck in processing large sets of texts. In the experiments, SNNTorch was able to slightly outperform the results obtained by other selected classifiers in the case of two PubMed datasets.

REFERENCES

- [1] J. Weissbock, A. A. Esmin, and D. Inkpen, "Using external information for classifying tweets," in *2013 Brazilian Conference on Intelligent Systems*, 2013, pp. 1-5.
- [2] M. Kozłowski and H. Rybinski, "Clustering of semantically enriched short texts," *Journal of Intelligent Information Systems*, vol. 53, no. 1, pp. 69-92, 2019.
- [3] I. Laña, J. L. Lobo, E. Capecci, J. Del Ser, and N. Kasabov, "Adaptive long-term traffic state estimation with evolving spiking neural networks," *Transportation Research Part C: Emerging Technologies*, vol. 101, pp. 126 - 144, 2019.
- [4] P. S. Maciąg, N. Kasabov, M. Kryszkiewicz, and R. Bembienik, "Air pollution prediction with clustering-based ensemble of evolving spiking neural networks and a case study for london area," *Environmental Modelling & Software*, vol. 118, pp. 262 - 280, 2019.
- [5] P. S. Maciąg, M. Kryszkiewicz, and R. Bembienik, "Online evolving spiking neural networks for incremental air pollution prediction," in *2020 International Joint Conference on Neural Networks (IJCNN)*, 2020, pp. 1-8.
- [6] H. Liu, G. Lu, Y. Wang, and N. Kasabov, "Evolving spiking neural network model for PM2.5 hourly concentration prediction based on seasonal differences: A case study on data from beijing and shanghai," *Aerosol and Air Quality Research*, vol. 21, no. 2, p. 200247, 2021.
- [7] L. Paulun, A. Wendt, and N. Kasabov, "A retinotopic spiking neural network system for accurate recognition of moving objects using neuCube and dynamic vision sensors," *Frontiers in Computational Neuroscience*, vol. 12, p. 42, 2018.
- [8] P. S. Maciąg, M. Kryszkiewicz, R. Bembienik, J. L. Lobo, and J. Del Ser, "Unsupervised anomaly detection in stream data with online evolving spiking neural networks," *Neural Networks*, vol. 139, pp. 118-139, 2021.
- [9] K. Demertzis and L. Iliadis, "A hybrid network anomaly and intrusion detection approach based on evolving spiking neural network classification," in *E-Democracy, Security, Privacy and Trust in a Digital World*, A. B. Sideridis, Z. Kardasiadou, C. P. Yialouris, and V. Zorkadis, Eds. Cham: Springer International Publishing, 2014, pp. 11-23.
- [10] N. K. Kasabov, "Neucube: A spiking neural network architecture for mapping, learning and understanding of spatio-temporal brain data," *Neural Networks*, vol. 52, pp. 62-76, Apr. 2014.

- [11] N. Kasabov and E. Capecchi, "Spiking neural network methodology for modelling, classification and understanding of eeg spatio-temporal data measuring cognitive processes," *Information Sciences*, vol. 294, pp. 565 – 575, 2015, innovative Applications of Artificial Neural Networks in Engineering.
- [12] J. K. Eshraghian, M. Ward, E. Neftci, X. Wang, G. Lenz, G. Dwivedi, M. Bennamoun, D. S. Jeong, and W. D. Lu, "Training spiking neural networks using lessons from deep learning," *arXiv preprint arXiv:2109.12894*, 2021.
- [13] J. L. Lobo, I. Oregi, A. Bifet, and J. Del Ser, "Exploiting the stimuli encoding scheme of evolving spiking neural networks for stream learning," *Neural Networks*, vol. 123, pp. 118 – 133, 2020.
- [14] Q. Li, H. Peng, J. Li, C. Xia, R. Yang, L. Sun, P. S. Yu, and L. He, "A survey on text classification: From traditional to deep learning," vol. 13, no. 2, apr 2022. [Online]. Available: <https://doi.org/10.1145/3495162>
- [15] J. Y. Lee and F. Dernoncourt, "Sequential short-text classification with recurrent and convolutional neural networks," *arXiv preprint arXiv:1603.03827*, 2016.
- [16] Chen, Yahui, "Convolutional neural network for sentence classification," Master's thesis, 2015. [Online]. Available: <http://hdl.handle.net/10012/9592>
- [17] Y. Hu, Y. Li, T. Yang, and Q. Pan, "Short text classification with a convolutional neural networks based method," in *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*. IEEE, 2018, pp. 1432–1435.
- [18] L. Yao, C. Mao, and Y. Luo, "Graph convolutional networks for text classification," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, pp. 7370–7377, Jul. 2019.
- [19] K. Kowsari, K. Jafari Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown, "Text classification algorithms: A survey," *Information*, vol. 10, no. 4, p. 150, 2019.
- [20] M. Białas, M. M. Mirończuk, and J. Mańdziuk, "Biologically plausible learning of text representation with spiking neural networks," in *Parallel Problem Solving from Nature – PPSN XVI*, T. Bäck, M. Preuss, A. Deutz, H. Wang, C. Doerr, M. Emmerich, and H. Trautmann, Eds. Cham: Springer International Publishing, 2020, pp. 433–447.
- [21] Y. Wang, Y. Zeng, J. Tang, and B. Xu, "Biological neuron coding inspired binary word embeddings," *Cognitive Computation*, vol. 11, no. 5, pp. 676–684, 2019.
- [22] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.
- [23] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *International conference on machine learning*, 2014, pp. 1188–1196.
- [24] J. L. Lobo, I. Laña, J. Del Ser, M. N. Bilbao, and N. Kasabov, "Evolving spiking neural networks for online learning over drifting data streams," *Neural Networks*, vol. 108, pp. 1 – 19, 2018.
- [25] B. Petro, N. Kasabov, and R. M. Kiss, "Selection and optimization of temporal spike encoding methods for spiking neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–13, 2019.
- [26] E. M. Izhikevich and G. M. Edelman, "Large-scale model of mammalian thalamocortical systems," *Proceedings of the National Academy of Sciences*, vol. 105, no. 9, pp. 3593–3598, 2008.
- [27] F. Ponulak and A. Kasiński, "Supervised learning in spiking neural networks with resume: sequence learning, classification, and spike shifting," *Neural computation*, vol. 22, no. 2, pp. 467–510, 2010.
- [28] F. Ponulak and A. Kasinski, "Introduction to spiking neural networks: Information processing, learning and applications." *Acta neurobiologiae experimentalis*, vol. 71, no. 4, pp. 409–433, 2011.
- [29] I. Feinerer, "Introduction to the tm package text mining in R," *Avail. on line*: <http://cran.r-project.org/web/packages/tm/vignettes/tm.pdf>, 2013.
- [30] R. Řehůřek and P. Sojka, "Software Framework for Topic Modelling with Large Corpora," in *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. Valletta, Malta: ELRA, May 2010, pp. 45–50, <http://is.muni.cz/publication/884893/en>.
- [31] BioASQ Team. (2021) A challenge in large-scale biomedical semantic indexing and question answering. [Online]. Available: <http://www.bioasq.org/participate/challenges>

Rule-based approximation of black-box classifiers for tabular data to generate global and local explanations

Cezary Maszczyk^{*†}, Michał Kozielski[‡] and Marek Sikora^{†‡}

^{*}Doctoral School, Silesian Univeristy of Technology,
ul. Akademicka 2A, 44-100 Gliwice, Poland

[†]Łukasiewicz Research Network – Institute of Innovative Technologies EMAG,
ul. Leopolda 31, 40-189 Katowice, Poland

[‡]Department of Computer Networks and Systems, Silesian Univeristy of Technology,
ul. Akademicka 16, 44-100 Gliwice, Poland
Email: marek.sikora@polsl.pl

Abstract—The need to understand the decision bases of artificial intelligence methods is becoming widespread. One method to obtain explanations of machine learning models and their decisions is the approximation of a complex model treated as a black box by an interpretable rule-based model. Such an approach allows detailed and understandable explanations to be generated from the elementary conditions contained in the rule premises. However, there is a lack of research on the evaluation of such an approximation and the influence of the parameters of the rule-based approximator. In this work, a rule-based approximation of complex classifier for tabular data is evaluated. Moreover, it was investigated how selected measures of rule quality affect the approximation. The obtained results show what quality of approximation can be expected and indicate which measure of rule quality is worth using in such application.

I. INTRODUCTION

IN RECENT years, eXplainable Artificial Intelligence (XAI) [1], [2], [3], [4] has become an increasingly important field of artificial intelligence (AI). The explanations generated by XAI methods allow people to understand how artificial intelligence methods work and are useful for many types of users involved in different ways with AI applications.

This work focuses on XAI derived from rule induction [5]. Rule-based models represent knowledge embedded in data in the form of IF-THEN rules. The premise of a rule is a conjunction of elementary conditions $w_i \equiv a_i \odot x_i$, with x_i , being an element of the attribute a_i domain and \odot representing a relation ($=$ for symbolic attributes; $<$, \leq , $>$, \geq for ordinal and numerical ones). The conclusion of the rule contains a decision, which can be either symbolic or numeric.

Rules and decision trees (which can be easily transformed into mutually disjoint rules) are used to generate global explanations because of their interpretability [4]. Such global explanations are obtained by generating a rule-based interpretable

This work was partially supported by: Computer Networks and Systems Department at Silesian University of Technology within the statutory research project and the Łukasiewicz Research Network (ROLAP-ML, R&D grant)

model that approximates the non-interpretable (complex) base model. The use of rule-based approximation is justified for tabular data where attributes are interpretable and the generated rules can be understood by humans.

There are recent examples in the literature of the use of rule-based systems to obtain model agnostic explanations. Rule-based models approximating the complex base model locally were considered in [6]. The works [7], [8] considered rules as interpretable expressions used to present explanations within a proposed local-to-global approach for black-box explanations.

The parameters of the rule-based model induction method affect the quality of the base model approximation. Furthermore, they affect the explanations generated from the set of input rules. To the best of the authors' knowledge, there is no published analysis showing how well a rule-based model can approximate the complex base model being explained. Therefore, the aim of this paper is to present and evaluate the rule-based, model-agnostic approach to explaining machine learning models. The rule-based approximation of a black-box model may be evaluated in terms of its accuracy and number of the generated rules. Therefore, three measures of rule quality used in the induction process are verified in this work: Correlation, C2 and Precision [9]. These measures, in the above order, generate more and more specific rules and thus build probably more and more accurate approximators but composed of an increasing number of rules. Verification of these parameters should provide clues that are useful in the development of rule-based approximators used to create explanations.

The contribution of this paper includes: (i) analysis and evaluation of the approach that generates global and local explanations from a rule-based approximation of a black-box model, (ii) verification of how the application of different measures of rule quality assessment (Correlation, C2 and Precision), which enable variation in the approximation accuracy of a machine learning model, affects the explanations obtained.

II. THE IDEA OF RULE-BASED EXPLANATIONS

The proposed approach to generating machine learning model explanations involves approximating the base model treated as a black-box using a rule-based model. Base model approximation means that the rule-based model learns the decisions of the base model. Such a learning process requires transformed training data in which the existing decision variable is replaced by the decision of the base model. In the extreme case (which may mean over-fitting the base model), the transformed training data may be identical to the original.

The generated rule-based model, although interpretable in theory, is not always clear, e.g. due to the number of rules. However, once the rule-based model is generated, the elementary conditions of the rules and their importance can be extracted. On their basis, it is possible to generate the explanations showing which attributes, and which attribute value ranges, are most important for decision making by the base model. Importance based rankings of elementary conditions are generated using the Shapley index [10] and details on the assessment of the importance of rule elementary conditions are presented in the work [11].

Using the above method to explain the black-box classifier, global and local explanations can be obtained. The global explanation of the classifier is in the form of a ranking of elementary rule conditions indicating which ones have the highest importance in the decision of the base model about the selected class. The local explanation, on the other hand, consists of a set of rules covering the data instance and an analogous ranking indicating the importance of the elementary conditions in the classifier's decision.

III. EVALUATION OF RULE-BASED APPROXIMATION

The first step in the presented concept of generating rule-based explanations of the base machine learning model is to approximate this model with a rule-based model. The base model approximation should be of high accuracy to generate reliable explanations. The rule induction algorithm can be controlled by a number of parameters. This study focuses on the rule quality measure used and verifies impact of three such measures: Correlation, C2 and Precision.

The approach presented in this paper was implemented in Python. The scikit-learn package [12] was used to generate base models and the coverage algorithm implementation available in the RuleKit package [13] via a Python wrapper¹ was used to generate the rule-based models. Four different algorithms were selected to generate complex, non-interpretable classifiers used as base models. The selected algorithms are: Artificial Neural Network (NN), Random Forest (RF), Support Vector Machine (SVM) and Extreme Gradient Boosting (XGBoost). The base models and the approximating rule-based models were evaluated in 10-fold cross validation process. Experiments were conducted on a set of 29 data sets available in the OpenML² and UCI³ repositories. These data sets consist

of tabular data and relate to the task of classification.

In the experiments, the quality of the implemented approximation with the rule-based model was evaluated in two ways. Firstly, the classification quality measures, such as Accuracy (Acc) and Balanced Accuracy (BAcc), were calculated. The quality of the approximator was determined for the transformed data in which the decision attribute represents the decision of the base model. Table I presents mean values calculated for all the data sets that were divided into train and test data. In the experiment, four base models were approximated and three rule quality measures were verified as values of the approximator parameter. Additionally, Table I presents the average number of rules that were generated by the induction algorithm.

TABLE I
QUALITY OF RULE-BASED APPROXIMATIONS ON TRANSFORMED TRAINING AND TEST DATA, FOR WHICH THE DECISION ATTRIBUTE IS THE DECISION OF THE BASE MODEL

Base model	Approx. param.	#Rules	Train data		Test data	
			Acc	BAcc	Acc	BAcc
NN	Correlation	19.9	0.938	0.932	0.894	0.878
	C2	34.2	0.968	0.965	0.916	0.902
	Precision	75.3	0.988	0.986	0.912	0.892
RF	Correlation	21.1	0.924	0.92	0.873	0.858
	C2	38.3	0.958	0.953	0.899	0.884
	Precision	82.5	0.979	0.976	0.894	0.877
SVM	Correlation	18.7	0.952	0.949	0.915	0.905
	C2	30.4	0.978	0.976	0.935	0.922
	Precision	67.1	0.989	0.986	0.938	0.922
XGB	Correlation	21.1	0.924	0.921	0.877	0.863
	C2	38.4	0.958	0.953	0.9	0.886
	Precision	81.1	0.98	0.977	0.893	0.877

Based on the results in Table I, it can be concluded that the rule-based models approximate complex models well. Furthermore, the results show that the accuracy of this approximation decreases significantly on the test data compared to the training data. This is an expected result, as rule-based models were generated to obtain the best possible approximation on the training data.

Besides, the results in Table I show that the selected measures - in order: Correlation, C2 and Precision - enable to obtain increasingly fit rule-based models. The increase in quality results from the increase in the number of rules. In other words, the more general the rules are the less accurate the approximation becomes. On test data, however, the best quality results are obtained by models generated using the C2 measure.

The generated base models and their approximators can be further evaluated on the original data sets. Fig. 1 presents the difference in Balanced Accuracy (BAcc) between the generated approximations and base models. The BAcc values were calculated on test data for each of the data sets. XGBoost was used as a base model and the C2 measure was used in rule-based approximation generation.

¹<https://github.com/adaa-pols/RuleKit-python>

²<https://openml.org/>

³<https://archive.ics.uci.edu/ml/datasets.php>

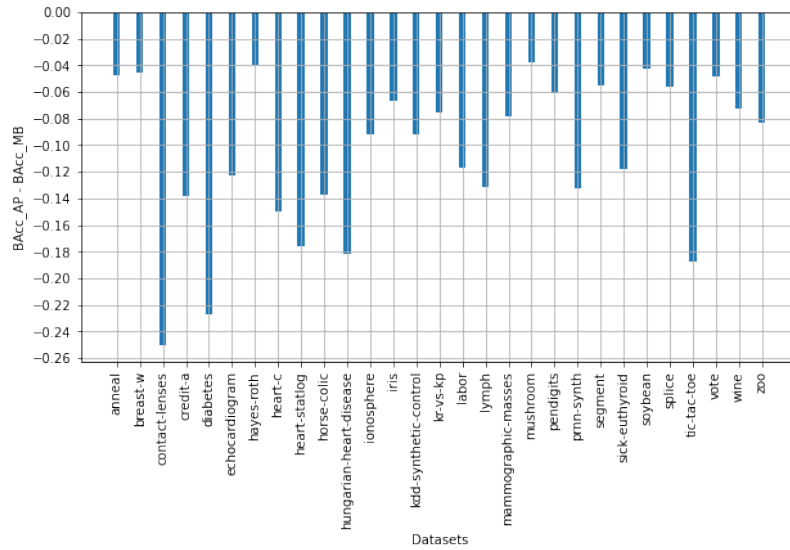


Fig. 1. Approximation quality illustrated as the difference of the Balanced Accuracy values for rule-based approximation (BACC_{AP}) and base model (BACC_{MB}) - comparison on test data

The results illustrated in Figure 1 show that an approximator operating on the original data is never better than the base model. These results provide a negative answer to the question of whether creating a rule-based model that cannot distinguish between examples that even complex models cannot correctly classify will produce rule sets with good classification quality. Moreover, it is again apparent that over-fitting to the train data results in lower accuracy on test data.

Within the second approach to rule-based approximation evaluation the generated explanations were compared. Therefore, using the SHAP method, attribute rankings were generated for the base models and their approximations. If the size of the data set exceeded 100 instances a subset of that size was selected to determine the Shapley values. Next, the top three features from both rankings were compared to verify if these three most important attributes were the same for both models. The results of the comparison are illustrated in Fig. 2. It presents for each data set the average number of identical features selected as the top three. This average is calculated as four base models (NN, Random Forest, SVM and XGBoost) were generated again for each data set. The rule-based approximation was performed using the C2 measure, which proved to be the most reasonable choice in earlier experiments. The comparison was performed on 31 data sets.

The results presented in Fig. 2 show that in most cases at least two out of three most important features are the same for the base model and its approximator.

IV. EXEMPLARY USE CASES

Having the rule-based approximator of sufficient accuracy induced it is possible to generate explanations. The example global and local explanations presented below were generated

from the rule-based model using RuleXAI package⁴. This package generates explanations by analysing the importance of the rules' elementary conditions [11]. The C2 measure was used for rule quality evaluation. The rule-based model approximated the XGBoost classifier.

The base model was trained on the wine⁵ data set representing a three-class problem and consisting of the following attributes: Alcohol, Malic_acid, Ash, Alkalinity_of_ash, Magnesium, Total_phenols, Flavanoids, Non-flavanoid_phenols, Proanthocyanins, Color_intensity, Hue, OD280_OD315_of_diluted_wines, Proline, Class.

The generated global explanations are presented in Table II. They consist of rankings built separately for each class. The rankings were built on the basis of feature importance, or more precisely on the basis of the importance of the elementary conditions of the rules.

Local explanations are generated for a specific data instance. The analysed data instance takes the following values for the above list of attributes {0.213157895, 0.029644269, 0.652406417, 0.381443299, 0.260869565, 0.420689655, 0.394514768, 0.169811321, 0.611987382, 0.151023891, 0.25203252, 0.663003663, 0.172610556, 1}. This data instance was covered by the following two rules from the approximating model:

```
IF Color_intensity = (-inf, 0.19) THEN
ref_prediction = 1.0
IF Alcohol = (-inf, 0.39) AND Flavanoids =
<0.13, inf) THEN ref_prediction = 1.0
```

In addition to the explanation in the form of rules, RuleXAI generated a ranking presented in Fig. 3. It shows which ranges of attribute values were most important for a given decision.

⁴<https://github.com/adaa-pols/RuleXAI>

⁵<https://archive.ics.uci.edu/ml/datasets/wine>

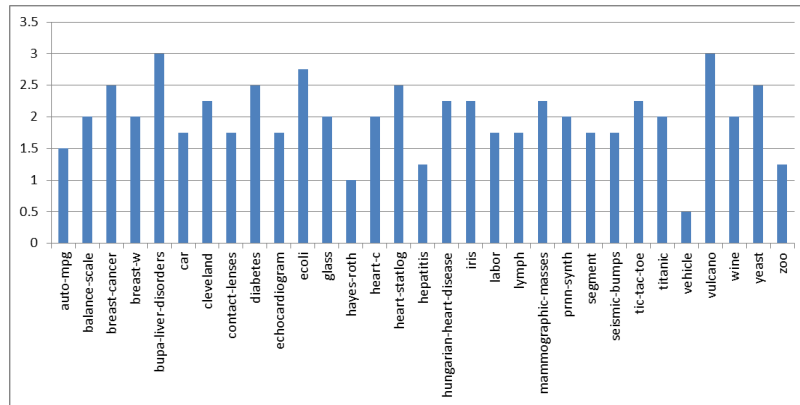


Fig. 2. Number of the same features in the first three positions of the rankings generated by the SHAP method for the base model and its approximator

TABLE II
RULE-BASED GLOBAL EXPLANATION GENERATED BY RULEXAI - THE EXPLANATION OF THE XGBOOST MODEL GENERATED ON WINE DATA SET

Class 0		Class 1		Class 2	
Condition	Importance	Condition	Importance	Condition	Importance
Proline ≥ 0.51	0.872	Color_intensity < 0.19	0.906	Flavanoids < 0.13	0.841
Proline ≥ 0.4	0.609	Alcohol < 0.39	0.728	OD280_OD315_of_diluted_wines < 0.2	0.825
Alcohol ≥ 0.52	0.352	Flavanoids ≥ 0.13	0.147	Ash ≥ 0.37	0.05
				Malic_acid ≥ 0.074	0.034

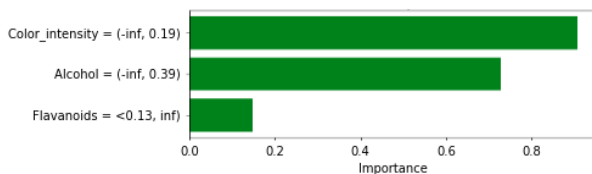


Fig. 3. Rule-based local explanation showing which attributes and which ranges of attribute values were most important for a given decision

V. CONCLUSIONS

The method generating global and local explanations using rule-based approximation of the black-box model was analysed within the presented research. In addition, global and local explanations generated from the rule-based approximation were presented. The explanations generated from the rules provide a detailed understanding of which features in which value ranges are important to the classifier's decisions.

Application of the rule-based model to approximation of the complex classifier generated on tabular data was positively evaluated. The approximation obtained with the rule-based models was consistent with the approximated base models both from the point of view of classification quality measures and from the perspective of consistency in the most important features indicated by the generated explanations.

REFERENCES

- [1] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi, "A survey of methods for explaining black box models," *ACM Comput. Surv.*, vol. 51, no. 5, aug 2018. [Online]. Available: <https://doi.org/10.1145/3236009>
- [2] A. Adadi and M. Berrada, "Peeking inside the black-box: A survey on explainable artificial intelligence (xai)," *IEEE Access*, vol. 6, pp. 52 138–52 160, 2018.
- [3] C. Molnar, *Interpretable Machine Learning*, 2nd ed., 2022. [Online]. Available: <https://christophm.github.io/interpretable-ml-book>
- [4] P. Biecek and T. Burzykowski, *Explanatory model analysis: Explore, explain and examine predictive models*. Chapman and Hall/CRC, 2021.
- [5] J. W. Grzymala-Busse, *Rule Induction*. Boston, MA: Springer US, 2005, pp. 277–294. [Online]. Available: https://doi.org/10.1007/0-387-25465-X_13
- [6] E. Pastor and E. Baralis, "Explaining black box models by means of local rules," in *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, ser. SAC '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 510–517. [Online]. Available: <https://doi.org/10.1145/3297280.3297328>
- [7] D. Pedreschi, F. Giannotti, R. Guidotti, A. Monreale, S. Ruggieri, and F. Turini, "Meaningful explanations of black box ai decision systems," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 9780–9784.
- [8] M. Setzu, R. Guidotti, A. Monreale, and F. Turini, "Global explanations with local scoring," in *Machine Learning and Knowledge Discovery in Databases*, P. Cellier and K. Driessens, Eds. Cham: Springer International Publishing, 2020, pp. 159–171.
- [9] M. Sikora and Ł. Wróbel, "Data-driven adaptive selection of rule quality measures for improving rule induction and filtration algorithms," *International Journal of General Systems*, vol. 42, no. 6, pp. 594–613, 2013.
- [10] L. S. Shapley, "A value for n-person games," *Classics in game theory*, vol. 69, 1997.
- [11] M. Sikora, "Redefinition of decision rules based on the importance of elementary conditions evaluation," *Fundamenta Informaticae*, vol. 123, no. 2, pp. 171–197, 2013.
- [12] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in python," *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.
- [13] A. Gudyś, M. Sikora, and Łukasz Wróbel, "Rulekit: A comprehensive suite for rule-based learning," *Knowledge-Based Systems*, vol. 194, p. 105480, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0950705120300046>

Automatic detection of potential customers by opinion mining and intelligent agents

Raúl Moreno
Madox Viajes
Rey Juan Carlos University
Arroyomolinos (Madrid), Spain
r.morenoi@alumnos.urjc.es

Alberto Fernández-Isabel
Rey Juan Carlos University
Data Science Laboratory
Móstoles (Madrid), Spain
alberto.fernandez.isabel@urjc.es

Isaac Martín de Diego
Rey Juan Carlos University
Data Science Laboratory
Móstoles (Madrid), Spain
isaac.martin@urjc.es

Javier M. Moguerza
Rey Juan Carlos University
Data Science Laboratory
Móstoles (Madrid), Spain
javier.moguerza@urjc.es

Carmen Lancho
Rey Juan Carlos University
Data Science Laboratory
Móstoles (Madrid), Spain
carmen.lancho@urjc.es

Marina Cuesta
Rey Juan Carlos University
Data Science Laboratory
Móstoles (Madrid), Spain
marina.cuesta@urjc.es

Abstract—Customer acquisition is an issue that continues to receive attention from companies worldwide. Various marketing campaigns using psychological methodologies have been designed to address this issue. However, once a campaign is launched, it is highly complicated to detect which sets of customers are most likely to purchase an offered product. This fact is key since it allows companies to focus their efforts on specific clients and discard others. Several selection techniques have been implemented, but most of them are usually very demanding in terms of time and human resources for the companies. Artificial Intelligence techniques appear to help to simplify the process. Thus, companies have started to use Machine Learning (ML) models trained to efficiently detect those clients with certain proneness to purchase. Toward this goal, this paper presents a novel purchase propensity detection ML system based on Sentiment Analysis techniques able to consider customer comments regarding the offered products. The tourist domain was selected for the case study, where the obtained product was successfully embedded in an initial prototype.

I. INTRODUCTION

CUSTOMER acquisition is one of the most important tasks companies undertake to promote their products and services. It allows expanding the business, enhancing their potential in the markets and producing benefits over time. However, companies need a clear a solid plan of action to be successful in custom acquisition. Creating such a plan in turn requires first-hand knowledge of customers' needs and tastes [1].

Several studies have been conducted to detect these customers' needs, producing behavioral profiles and other different psychological approaches. These proposals work acceptably when companies have specific employees or departments dedicated to making evaluations of the customers. In the case of small or medium enterprises with few resources, or those with high amounts of customers to evaluate, this task is difficult to achieve, becoming a very demanding process.

This work was supported by Madox Viajes Travel Agency

On the other hand, the Internet makes it easy to access collected information about customers' opinions on purchases. Thus, these data can be analyzed to detect patterns that indicate a certain propensity to buy some specific products offered by a company [2]. Specifically, customers whose search is more alternative-based are found to have a higher propensity to purchase than customers whose search is more attribute-based. The main rationale is that customers using alternative based search are evaluating products one at a time, and they are more able to judge whether the product meets their purchasing criteria or it is suitable for them with lesser distraction from other products or services.

In the case of the tourism domain, it is very difficult to build client loyalty due to the wide amount of offers and the fluctuations in the market. Moreover, the recent COVID-19 outbreak has provoked more complications for tourist companies due to the restrictions and measures imposed by governments of countries all over the world. As a result, the acquisition of customers (whose numbers reduced drastically during the period) and the detection of those who are prone to purchase has become a fundamental issue, especially for small or medium-sized companies. Due to this situation, this process is also more demanding and time-consuming, so it is important to develop specific software to support the decision-making step and provide recommendations automatically.

This paper presents a Machine Learning (ML) module able to automatize the detection of potential customers. It has been successfully embedded into an initial prototype of a tourist management framework. The fusion of both elements (i.e., the tourist domain and ML) has produced a novel and complete system based on expert knowledge to manage tourist events and provide recommendations from the perspective of the customer and the travel agency. This enhancement allows the tour companies to reduce efforts and focus their resources on specific individuals and tourist offers.

Regarding the ML module, it analyzes the opinions in texts written by potential customers for different tourist offers, being able to detect the real interest of the individuals. This allows discarding those customers with less predicted interest. Intelligent agents following a Multi-Agent System (MAS) architecture have been included to establish communication and knowledge interchange. This MAS promotes the distribution of the workload and eases the evaluation of multiple texts. This issue is relevant because companies usually need real-time analysis and responses to satisfy the requests made by customers.

Several experiments in real environments have been implemented to show the viability of the proposal. *Madox Viajes* is a tour company that has put in production the developed software, including the ML module with very successful and promising results.

The rest of the paper is organized as follows. Section II situates the approach in the domain and makes some comparisons with similar ideas. Section III details the architectures of the ML module and the prototype. Section IV illustrates a battery of experiments achieved in a real environment. Finally, Section V concludes by making a detailed analysis and provides future guidelines.

II. BACKGROUND

This section introduces the foundations of the ML module specifically designed to evaluate the opinions of customers. Opinion Mining techniques are put into the spotlight for the case of evaluating opinions. Then, intelligent agents and MAS are detailed focusing on the distribution of the workload. Finally, a first analysis addresses the problematic in tourism related to the detection of prone to purchase customers (see Section II-C).

A. Sentiment Analysis for the evaluating opinions

Sentiment Analysis (also called Opinion Mining) is one of the most relevant areas in the Natural Language Processing (NLP) domain. Its main objective is to gather the subjectivity expressed by humans in texts [3]. Thus, it measures the influence of a written text over the reader regarding the type of feelings risen and the level of affection.

There are two main perspectives in NLP to address the Sentiment Analysis issue: dictionary-based approaches and ML approaches.

Dictionary-based approaches (also called lexicons) consist of a collection of predefined words which usually appear associated with a sentiment score or a polarity (positive, negative or neutral). They are used to calculate the sentiment polarity of a sentence detecting words included in the dictionary and averaging the polarity of these words. It can be also extrapolated to paragraph or complete texts. However, it has to be considered the noise generated by the set of relevant words (e.g., substantives, verbs or adjectives) that are not detected, and the changes in the discourse made by the creator of the textual content. Well-known examples of these dictionaries are SenticNet [4] and SentiWordNet [5].

ML approaches predict the sentiment values of the words by using statistical models based on distributional semantics (e.g., word embeddings and transformers). These models are built through a training phase on which a collection of words with their corresponding polarity is used. This collection is a corpus that contains the ground truth (i.e., the reality that ML models must try to simulate adapting their parameters to learn). Subsequently, two more steps are usually included: the validation and the test phases. Both allow measuring the quality of the model regarding the ground truth (i.e., the learning capability).

Delving into the ML solutions, there exist several types of approaches to address the Sentiment Analysis task [6]. For instance, there are simple solutions that only use a basic ML discarding the processing step [7]. Other approaches use NLP techniques at the beginning of the pipeline and different embedded ML models later to produce a more complex model [8]. In the first ones, Deep Learning approaches are usually the most typical ones. They do not consider rendering the textual context as they are focused on detecting the syntax and semantics patterns of the text. Common instances of these models are word embedding-based approaches [9] as Word2Vec and LSTM neural networks. However, their performance cannot be compared to the quality provided by the most recent models implemented by bidirectional transformers and attention methods (e.g., BERT [10] and their related approaches). These ML models usually include two attention methods to estimate the value of their parameters: intra-attention and global attention [11]. The first estimates the similarity between words in a sentence, while the second follows a global perspective taking into account the whole textual content.

In the case of the Sentiment Analysis focused on opinions and reviews, the approaches are usually ML solutions adapted to the specific context. However, these proposals are usually limited to only make an estimation and later their results are complemented by the knowledge of human experts. Typical instances of this perspective are: course evaluations [12], online purchase evaluations [13] and movie reviews [14].

The proposed ML module achieves the Sentiment Analysis task, and later gathering conclusions from the results through the distributed intelligence provided by the intelligent agents. Thus, the tourism domain prototype where it is embedded is able to predict which individuals are prone to purchase tourist offers and events.

B. Intelligent agents to distribute the workload

Intelligent agents are software abstractions able to simulate interactions (with an environment or with other agents) and behaviors from the real world. These elements are proactive, autonomous, and independent, addressing different problems according to the set of predefined rules and knowledge. Their ability to interact can be exploited to solve complex problems or to distribute the workload with certain coordination. Thus, MAS appear as a possible solution. These MAS are usually designed following a level of abstraction. Agent-Based Modeling (ABM) [15] and Agent-Oriented Software Engineering













Concept	Meaning	Icon
Agent	An active element with explicit goals that is able to initiate some actions involving other elements of the simulation.	
Role	A specific collection of tasks performed by agents when pursuing a goal or offering some service to the other members of the society. It has as a result a specific behavior.	
Environment Application	An element of the environment. Agents can act on the environment using its actions and perceive information through its events.	
Goal	An objective of a role/agent. Roles/agents try to satisfy their goals executing tasks. A goal is achieved or fails if some elements (i.e. frame facts and events) are present or absent in the agent groups or the environment.	
Task	A capability of a role/agent. To execute a task, certain elements (i.e. frame facts and events) must be available. The execution produces/consumes some elements as result.	
Frame Fact	An element produced by a task, and therefore by the roles/agents.	
Mental State	Part of the internal state of a role/agent. It groups goals, frame facts and events, and specify conditions on them.	
Belief	Part of the mental state. It contains the knowledge (specific, role-based or general) that an agent possesses. This knowledge can be used to interact with its surrounded environment.	
Conversation	Communication between two or more agents to exchange information. One of these agents plays the role of initiator of the conversation, while the others are the receptors.	
Plan	Representation of the means by which a goal can be satisfied. It is usually structured to provide a deterministic meaning to the operations.	
Mental State Manager	Part of the internal state of a role/agent. It provides for operations to create, destroy and modify mental entities.	
Mental State Processor	Part of the internal state of a role/agent. It determines how a mental state evolves, described in terms of rules or planning.	

Fig. 1. Main concepts of INGENIAS for developing MAS.

(AOSE) [16] provide the elements and entities to tackle this issue.

Intelligent agents present a life-cycle focused on satisfying a collection of goals through several associated concepts (see Fig. 1). These goals are structured in a hierarchical way, having sets of sub-goals that accomplish other goals at higher levels. Goals are associated with a set of tasks that are executed by the agents. Both goals and tasks are part of the mental state of the agents. This mental state plays the role of the brain, containing rules and specific knowledge from the environment that support the execution of the tasks and the satisfaction of the goals.

Agents can take advantage of their ability to interact with the environment to solve complex problems having partial or reduced knowledge about a problem. The organization in MAS opens the collaboration, the competition and also the negotiation. Well-known approaches that use MAS to solve complex problems or simulate real environments are road traffic simulations [17], distributed decision support systems [18], bio-inspired ML-based systems [19], and computer games

[20].

MAS are usually modeled using specific artifacts and entities to tackle the development steps of complex systems. It allows generating graphically relationships and interactions between agents and later transforming them to source code automatically. INGENIAS, GAIA, Prometheus, and Tropos are well-known agent modeling methodologies in charge of providing support through specific languages [21].

Regarding the implementation of MAS, agent platforms are the typical solution. These proposals include features to ease the distribution of the agents and manage their communication channels. Standing out approaches in this area are JADE [22] and MESA [23].

In the presented approach, the selected agent methodology has been INGENIAS, adapting the agent model to the MESA framework. It uses a bio-inspired distribution model based on the behavior of ant colonies that produce a MAS organized in several cumuli of agents (anthills) working together to achieve the Sentiment Analysis task and the evaluation of customers to detect who are prone to purchase.

C. Customers in tourism

Tourism is one of the most important economic activities all over the world. Everyday, million people are traveling to different destinations in a wide bunch of means of transport: airplanes, trains, cars, etc. This movement of people is usually related to work activities and tourist events. In the first case, individuals manage their travels by themselves (e.g., commuters), while in the second, it is usually addressed by tour companies (e.g., holidays and honeymoons). These companies provide counseling and support to the customers, being responsible of managing and coordinating the different steps during the travel and their stay [24].

Tourism is a volatile active, where several changes in the prices and in the market affect the benefits of the companies, as well as the configurations of travels and its features. World events like the COVID-19 outbreak or the Ukrainian war are recent situations that have produced hard modifications in the tourist sector. For this reason, the identification and capture of new customers, and the detection of those ones that are prone to purchase is basic for companies to optimize their workload.

Delving into the proneness of customers to purchase a tourist activity, a wide range of variables motivates and make some particular purchase decisions [25]. Instances of the most typical variables are: the particular culture, the emotional and physical state, and personal issues (e.g., visit a friend or a relate). These variables are called as motivators and can be classified into six main categories: primary motives, secondary motives, rational motives, emotional motives, conscious motives, and dormant motives [26].

Primary motives are those situations that force a person to purchase a tourist offer (e.g., a health problem with a relative who is in another country), and secondary motives consists of situations that modify the choice of the customers (e.g., the price of a flight according to different companies). In the case of the rational and emotional motives, the first is

related to an objective evaluation of the customer (e.g., a group of several members rents a minibus instead of some particular vehicles), while the second is the opposite, the customers are moved by their feelings (e.g., a person is prone to purchase a flight with a company instead of another one due to only personal preferences). Conscious motives encompass those ones where customers are aware of their personal needs (e.g., customers have to travel to a place where there is not a train station, therefore renting a car could be a better option). Finally, dormant motives are those ones that are unconscious and usually related to the influence of the society over the customers (e.g., customers have to travel to a place where individuals living there usually have high incomes, then the customers prefer to rent a car instead of traveling by bus).

To address and detect these motivations, the knowledge of psychological experts and the experience provided by travel agents become very relevant. However, when a company has to evaluate hundreds of proposals each day, the human evaluation is almost impossible. Thus, different systems and models to mitigate the problem has been developed. Some of them are very specific in the tourist domain and consider only behavioral features [27]. Other approaches are focused on the impact of the current technologies in tourism and how to adapt the offers to the high connectivity world nowadays [28]. And finally, there are others that evaluate the obtained results of the companies avoiding to consider the opinion and the level of satisfaction of customers [29].

In the case of the presented approach, customers are put into the spotlight, evaluating their opinions before their purchase, knowing their proneness and preferences for the different offers proposed. It allows developing a recommendation process for customers, and it eases the work of the travel agencies being able to discard and focus on specific customers in a smart way.

III. PROPOSED FRAMEWORK

This section details the novel ML module specially designed to evaluate the opinions of potential customers regarding a set of possible tourist offers, and the initial prototype of the tourist framework where it is embedded.

The ML module allows companies to reduce the effort in time and human resources, as employees can focus only on the customers that are more prone to purchase a product. This functionality works by analyzing the textual comments provided by the customers through Sentiment Analysis techniques and MAS to distribute the workload of the system.

In the case of the tourism domain, the prototype consists of a system focused on the two main perspectives related to the tourism market: customers and travel agents. It provides some features to manage the roles independently covering all the processes from the customers' perspective and the travel agents' perspective. Moreover, it can organize the different interactions between users with both roles (e.g., a tourist offer managed by a travel agent and a customer both using the system). From the customers (i.e., the tourists), the system can make recommendations according to specific preferences

through similarity comparisons based on their profile, as well as modifications through filtering processes. On the other hand, from the travel agents, the system includes the tourist services and offers available during the day, the ratings, the feedback of former customers about the services, and a novel functionality to indicate the most prone to purchase customers.

Next section provides further details about the architecture of the ML module. Next, the design of the prototype is tackle focusing on the different modules of the system and highlighting how the ML module is included in the release.

A. Propensity to purchase detection ML module

The ML module is organized into two main components: the *module administrator* and the *Sentiment Analysis administrator*. Both components work together internally (see Fig. 2).

The first component is the manager of the complete module, being in charge of obtaining the stored opinions of customers, managing the *Sentiment Analysis administrator* component, and producing the final results from the outcome of the former component. It comprehends three different entities: the *texts gatherer*, the *workload manager* and the *result processor*.

The *texts gatherer* collects the opinion to be processed by the module. They are loaded from the *Services information* database. This element cleans the text and applies corrections to misspelled words.

The *workload manager* organizes the texts using a Kafka queue and checks if the independent anthills are busy. It can create new MAS on-demand automatically or implement load balancing politics. The configuration of the ML module consists of 4 anthills by default.

The *result processor* obtains the outcomes from the anthills and organizes them to be stored in the database. Thus, the opinion of the customers regarding a tourist offer or service is labeled as positive or negative. Then, the system can select those customers prone to purchase.

The second component is the container of the anthills. These latter are bio-inspired hierarchical structures organized to distribute the workload between several agents. Therefore, the system is ready for processing the information provided by a big amount of potential customers. These anthills are MAS formed by a queen and sets of soldiers and workers, organized in a similar way to a real ant family structure. The number of agents playing the roles of the soldiers and workers can be modified by the system, though they are prefixed to 5 and 10 respectively. The queen agent is fixed by default to one per anthill, and that fact cannot be modified in the present release of the framework. The component is completed with the previously trained ML model that encompasses techniques to achieve the Sentiment Analysis task.

Delving into the Sentiment Analysis model, initially, the Universal Sentence Encoder (USE) was used for generating embeddings from the customer comments. Then, a Convolutional Neural Network (CNN) with the next sequential model pipeline is used: an input 256 dense layer by using ReLU activation, a dropout layer with a rate of 0.5, a 128 dense

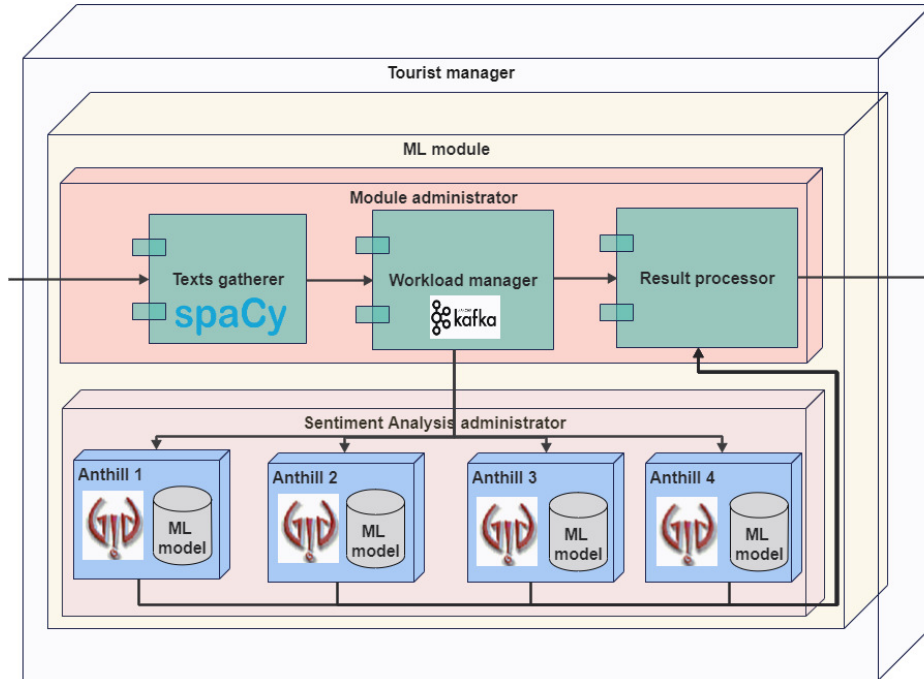


Fig. 2. Architecture of the ML module inside the *tourist manager* with the default configuration.

TABLE I
SUMMARY OF THE CNN CONFIGURATION.

Layer pipeline	Configuration
ReLU dense layer	256
Dropout	0.5
ReLU dense layer	128
Dropout	0.5
Softmax dense layer	2

layer by using also ReLU activation, another dropout layer with rate 0.5 and the end layer with a 2 dense layer by using a Softmax activation (see Table I).

Regarding the intelligent agents, they are the independent entities that process the texts through cooperative activities. A MAS consists of an anthill that encompasses three types of roles for the agents: queen, soldiers, and workers (see Fig. 3). Therefore, it is organized following bio-inspired ant social structure. The queen (i.e., the manager agent) is responsible for the anthill, receiving the texts to analyze directly from the *workload manager*. Notice that several texts can be sent to this agent. Then, the queen agent assigns the activity of evaluating the text to one soldier agent (i.e., the evaluator agent). This agent is in charge of assigning pieces of text to promote the distribution (usually paragraphs) or complete texts to the worker agents. These latter process the text through the Sentiment Analysis model provided to the anthill (each anthill has a copy of the ML model). Once worker agents have concluded their task, the soldiers join the texts if necessary (protecting and supervising the result obtained by a worker or a set of them) and they return the result to the queen agent.

This process follows the Belief-Desire-Intention (BDI) model [30] where each agent present a goal or a set of them to satisfy to complete its life-cycle. In this sense, the queen has the *assignedtask* goal, the soldiers have the *evaluatedopinion* goal and the workers have the *processedtext* goal. All the goals usually include associated tasks that are actions to achieve. Notice that the agents present at least one task that solves their corresponding goals. These tasks are applied in the environment that agents share. In this case, the environment is the current opinion or set of them of the possible customers. Each agent incorporates a mental state and a set of beliefs (motivations). The queen has in them simple rules to manage the texts, while soldiers present similar rules to distribute the text between the workers. However, workers include the ML model in their mental states to achieve the evaluation of the texts and some simple rules to organize the process in the beliefs. Finally, interactions between the individuals follow the hierarchical structure. They are completed through direct conversations. Notice that in this case, workers do not need to establish conversations with other workers since they tackle their commitment individually according to the orders of the soldier.

The design of the anthill model has been addressed through the INGENIAS agent methodology. Then, the resulting composition has been transformed to be compliant with the MESA framework. The conversations and interactions of the agents have been implemented following the Foundation for Intelligent Physical Agents (FIPA) standards [31].

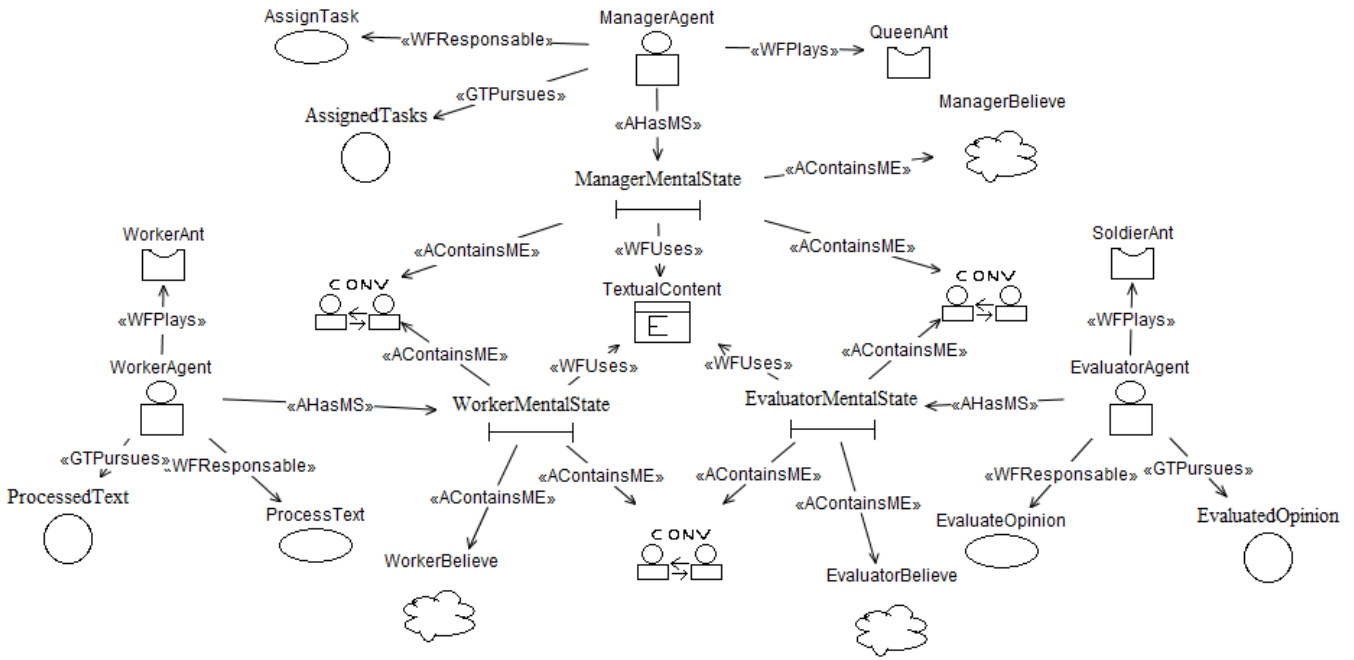


Fig. 3. Excerpt of the main entities involved in the anthill MAS.

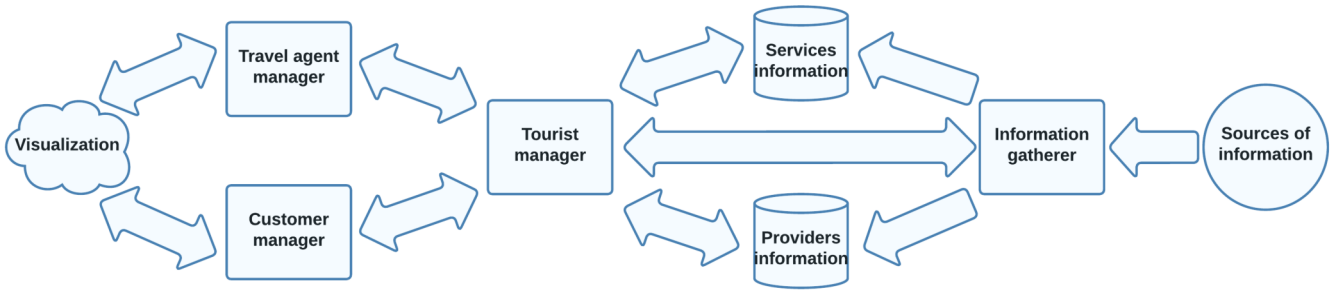


Fig. 4. Overview of the main architecture of the prototype.

B. Architecture of the prototype

The architecture of the prototype comprehends two databases and four main modules. These elements are related between them interacting automatically to recover information or responding the requests made by users through the graphical interface (see Fig. 4).

Regarding the databases, they are: *services information* and *providers information*. The first stores the information of the tourist services managed by the company. Both final users (i.e., independent customers and travel agents) produce relevant data that is stored here. It also contains a historic to generate different visual forms for the business. Note that information about customers and their opinions are also placed in that database. Therefore, it is used by the novel ML module to obtain textual content and store the obtained results. On the other hand, the second conserves the knowledge about the providers (i.e., the different companies that daily offer tourist services). That information is updated every day through an

automatic process but it needs the support of an expert in the domain to accomplish some specific and complex tasks related to the several offers and modes. The visual interface of the system provides specific graphical assistants and the architecture presents some modules to properly interact with both databases.

In the case of the main modules, they are: *information gatherer*, *tourist manager*, *travel agent manager* and *customer manager*. Notice that the first two modules are generic to every operation in the system, while the other two are specific for travel agents and customers respectively.

The *tourist manager* module is the core of the framework. It contains information about the opportunities and the profiles and preferences of the final users. With this information, it can produce recommendations to provide the best tourist resources for a trip. It is the module of the system that embeds the proposed ML module. It allows the system to select the most interesting customers according to the evaluation made through the analysis of their opinion.

The *information gatherer* module recapitulates and processes the information from the virtual tourism market. This market reflects the exchanges of information between the providers and tour companies. This module collects static and dynamic information. The first is information that is not frequently updated (e.g., the name of a hotel and its description). The second is volatile information that fluctuates several times during the day (e.g., occupancy levels or rates). Notice that some of this information cannot be stored in the system due to legal issues, so it is always consulted directly from web information sources.

The *customer manager* module is in charge of processing the events and interactions made by the customers. It provides information to the *tourist manager* module to collect information about a specific individual or tourist services and adequate offers depending on the preferences of the customers. Notice that the operations made by individuals with that role must be approved by a travel agent at some point in the pipeline before formalizing a possible trip.

The *travel agent* module provides assistance to travel agents to consult, modify and also create tourist offers. It also supports the selection of specific tourist resources to configure a trip. To achieve that task, this module makes use of the *tourist manager* module. Moreover, the filtering process of prone to purchase customers is also used here to show that knowledge to the different travel agents on demand.

IV. EXPERIMENTS

This section details the experiments achieved with the prototype of the framework in a real environment. This environment has been provided by *Madox Viajes*, a tour company that has implanted the current version of the system.

The experiment starts training the ML model used by the system. In this sense, a dataset provided by the company has been used. It consists of a set of 11,840 labeled observations being divided into 90% for training and 10% for testing purposes. These data is unbalanced since the most of opinions are from opportunities that were not bought by customers. Thus, the range of the texts in the dataset is 60% (not bought) and 40% (bought). The decision of using an unbalanced dataset is related to the idea of reducing the false positive predictions as in real environments as the customers are more likely not to purchase.

The output of the ML model is the probability of a tourist opportunity offered by the company of being purchased. Therefore, the company's employees could use this insight when working on opportunities. It should be noted that the focus on employees is important, as it requires really explaining to them what it means to "purchase" a tourist opportunity. In this case, it was necessary to make them understand that it is a probability that can guide them to focus on the best opportunities and discard or defer the rest. Thus, this probability was binarized to simplify this information to the company's employees. Several thresholds for the probability have been tested. The performance results are presented in Table II. In

this case, the optimal threshold was 0.1, corresponding to a F_2 -score equals to 0.8037, Precision 0.8488 and Recall 0.7932, respectively. That is, the ML model is right almost in the 85% of times when the prediction given by the model is "purchase". On the other hand, the ML model is able to recover more than the 79% of all the sold opportunities. Notice that the selection of the F_2 -score is motivated by the fact that for the company it is more relevant higher values of Recall to reduce efforts and increment the benefits.

After deploying the model in production, the ML module were embedded in the system with a default configuration of four anthills of intelligent agents. Then, the team of the company was trained in the use of the scoring produced by the ML model.

The system has been working during several months in the company. It was planned to measure the effect of scoring on the sales pipeline 8 months later (i.e., it was deployed on September 1st, 2021). Thus, the results were shown in the yearly harvest report for opportunities on April 1st, 2022 (see Table III). It is necessary to highlight that the 2021 sales were aligned with the 2020. It implies a normal behavior, as the company continued operating under pandemic circumstances. In this case, percentages over the total number of opportunities are used to avoid confidential data of the company that could provide insights to competitors. The most relevant information in order to evaluate the performance of the proposed ML module is the percentage of opportunities created in 2021 and sold in 2022: 2,78%. Notice that this is the highest percentage of opportunities sold in one year and were created in the previous year. This means that the scoring is really impacting positively on the travel agents. In fact, Table IV shows the percentages of opportunities created in a year and sold in the same year or in the next year. For instance, given the total amount of opportunities created in 2017 that were sold, the 89.2% were sold in the same year and the remaining ones (9.8%) were sold in 2018. In 2022, the effect of the ML model can be visualized. Thus, almost half of the tourist opportunities created in 2021 have been sold during the first four months of 2022 (41.9% which is almost more than 10 times increase compared to the previous year's value), being this number the highest in the history of the corresponding tourist company.

In conclusion, the incorporation of the proposed ML model in the framework has increased the detection of opportunities of interest for new customers. This fact has been translated into better business statistics, reducing the effort of the company in the creation of new packages, and the reduction of time demand for the employees.

V. CONCLUSIONS

This paper has presented a ML module created to evaluate the comments about different tourist offers and services, and to measure the propensity of potential customers to purchase them. It allows classifying the customers according to their proneness of completing a booking (i.e., it detects the most interesting customers). The module has been embedded in a prototype of a framework specifically designed for tourist

TABLE II
PERFORMANCE OF THE ML MODEL FOR DIFFERENT THRESHOLDS FOR THE PROBABILITY OF PURCHASE.

Metrics / Threshold	0.01	0.1	0.2	0.3	0.4	0.5
Precision	0.5476	0.8488	0.8920	0.9123	0.9299	0.9386
Recall	0.8639	0.7932	0.7660	0.7349	0.7141	0.6945
Accuracy	0.8892	0.9546	0.9574	0.9562	0.9557	0.9543
F₂-score	0.7744	0.8037	0.7882	0.7646	0.7488	0.7326

TABLE III
YEARLY HARVEST FOR PREDICTED PURCHASE AND PURCHASE OPPORTUNITIES ON APRIL 1st, 2022.

Year Created / Year Sold	2017	2018	2019	2020	2021	2022	Total
2017	14, 35%	1, 70%					16, 05%
2018		30, 94%	0, 72%				31, 66%
2019			35, 34%	0, 81%			36, 14%
2020				8, 43%	0, 36%		8, 79%
2021					3, 86%	2, 78%	6, 64%
2022						0, 72%	0, 72%
Total	14, 35%	32, 65%	36, 05%	9, 24%	4, 22%	3, 50%	100, 00%

TABLE IV
COMPARISON BETWEEN CREATED AND SOLD OPPORTUNITIES ON APRIL 1st, 2022.

	2018	2019	2020	2021	2022
Opportunities created and sold in the previous year	89.2%	97.7%	97.8%	95.6%	58.1%
Opportunities created in the previous year and sold in the current year	9.8%	2.3%	2.2%	4.4%	41.9%

management from both perspectives: the customer and the travel agent.

The ML module analyzes the opinion of the possible customers through Sentiment Analysis techniques based on neural networks. Moreover, it includes a bio-inspired architecture design of MAS to distribute the workload of the system. The complete system has been deployed in a real environment, showing its ability to manage a real tour company. The obtained results have been highly satisfying; the system has increased the ability of the company to find new potential customers. These results have translated into more economical benefits, simplification of the creation of new tourist opportunities, and less time spent by the human resources.

In the future, the MAS organization will be stressed with a test battery to find possible weaknesses of the architecture. Regarding the ML module in general, it will be improved by incorporating other techniques like Weight of Evidence (WoE). This fact will produce a novel and complete release of the system for the tourist domain. The architecture of the framework will be also improved and later adapted to be used through micro-services. All these upgrades will lead to a complete deployment where travel agents and customers will have access in real-time to the provided functionalities.

ACKNOWLEDGMENT

This research has been supported by the grants from Madrid Autonomous Community (Ref: IND2019/TIC-17194), the Spanish Ministry of Economy and Competitiveness, under the Retos-Investigación program: MODAS-IN (Ref: RTI-2018-094269-B-I00), and Rey Juan Carlos University (Ref: C1PREDOC2020); and donation of the Titan V GPU by NVIDIA Corporation.

REFERENCES

- [1] J. S. Thomas, "A methodology for linking customer acquisition to customer retention," *Journal of marketing research*, vol. 38, no. 2, pp. 262–268, 2001. doi: 10.1509/jmkr.38.2.262.18848
- [2] O. Mintz, I. S. Currim, and I. Jeliakov, "Information processing pattern and propensity to buy: An investigation of online point-of-purchase behavior," *Marketing Science*, vol. 32, no. 5, pp. 716–732, 2013. doi: 10.1287/mksc.2013.0790
- [3] K. Ravi and V. Ravi, "A survey on opinion mining and sentiment analysis: tasks, approaches and applications," *Knowledge-based systems*, vol. 89, pp. 14–46, 2015. doi: 10.1016/j.knosys.2015.06.015
- [4] E. Cambria, Y. Li, F. Z. Xing, S. Poria, and K. Kwok, "Sentinet 6: Ensemble application of symbolic and subsymbolic ai for sentiment analysis," in *Proceedings of the 29th ACM international conference on information & knowledge management*, 2020. doi: 10.1145/3340531.3412003 pp. 105–114.
- [5] N. Medagoda, S. Shanmuganathan, and J. Whalley, "Sentiment lexicon construction using sentiwordnet 3.0," in *2015 11th International Conference on Natural Computation (ICNC)*. IEEE, 2015. doi: 10.1109/ICNC.2015.7378094 pp. 802–807.
- [6] M. Ahmad, S. Aftab, S. S. Muhammad, and S. Ahmad, "Machine learning techniques for sentiment analysis: A review," *Int. J. Multidiscip. Sci. Eng.*, vol. 8, no. 3, p. 27, 2017. doi: 10.18090/samriddhi.v12i02.03
- [7] S. Bhattacharya, D. Sarkar, D. K. Kole, and P. Jana, "Recent trends in recommendation systems and sentiment analysis," *Advanced Data Mining Tools and Methods for Social Computing*, pp. 163–175, 2022. doi: 10.1016/B978-0-32-385708-6.00016-3
- [8] R. Prabowo and M. Thelwall, "Sentiment analysis: A combined approach," *Journal of Informetrics*, vol. 3, no. 2, pp. 143–157, 2009. doi: 10.1016/j.joi.2009.01.003
- [9] A. Onan, "Sentiment analysis on product reviews based on weighted word embeddings and deep neural networks," *Concurrency and Computation: Practice and Experience*, vol. 33, no. 23, p. e5909, 2021. doi: 10.1002/cpe.5909
- [10] M. Pota, M. Ventura, H. Fujita, and M. Esposito, "Multilingual evaluation of pre-processing for bert-based sentiment analysis of tweets," *Expert Systems with Applications*, vol. 181, p. 115119, 2021. doi: 10.1016/j.eswa.2021.115119
- [11] N. Liu, B. Shen, Z. Zhang, Z. Zhang, and K. Mi, "Attention-based sentiment reasoner for aspect-based sentiment analysis," *Human-centric Computing and Information Sciences*, vol. 9, no. 1, pp. 1–17, 2019. doi: 10.1186/s13673-019-0196-3

- [12] A. Onan, "Sentiment analysis on massive open online course evaluations: a text mining and deep learning approach," *Computer Applications in Engineering Education*, vol. 29, no. 3, pp. 572–589, 2021. doi: 10.1002/cae.22253
- [13] S. Wassan, X. Chen, T. Shen, M. Waqar, and N. Jhanjhi, "Amazon product sentiment analysis using machine learning techniques," *Revista Argentina de Clínica Psicológica*, vol. 30, no. 1, p. 695, 2021. doi: 10.24205/03276716.2020.2065
- [14] P. K. Mallick, P. Dutta, S. Mishra, and M. K. Mishra, "Sentiment analysis and evaluation of movie reviews using classifiers," in *Cognitive Informatics and Soft Computing*. Springer, 2021. doi: 10.1007/978-981-16-1056-1_5 pp. 53–59.
- [15] M. Uddin, Q. Wang, H. H. Wei, H. L. Chi, and M. Ni, "Building information modeling (bim), system dynamics (sd), and agent-based modeling (abm): Towards an integrated approach," *Ain Shams Engineering Journal*, vol. 12, no. 4, pp. 4261–4274, 2021. doi: 10.1016/j.asej.2021.04.015
- [16] A. Garro, M. Mühlhäuser, A. Tundis, M. Baldoni, C. Baroglio, F. Bergenti, P. Torroni *et al.*, "Intelligent agents: Multi-agent systems," *Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics*, vol. 315, 2018. doi: 10.1016/B978-0-12-809633-8.20328-2
- [17] A. Fernández-Isabel, R. Fuentes-Fernández, and I. M. de Diego, "Modeling multi-agent systems to simulate sensor-based smart roads," *Simulation Modelling Practice and Theory*, vol. 99, p. 101994, 2020. doi: 10.1016/j.simpat.2019.101994
- [18] C. González-Fernández, J. Cabezas, A. Fernández-Isabel, and I. Martín de Diego, "Combining multi-agent systems and subjective logic to develop decision support systems," in *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*. Springer, 2020. doi: 10.1007/978-3-030-50146-4_12 pp. 143–157.
- [19] J. Cabezas, A. Fernandez-Isabel, R. R. Fernández, C. González-Fernández, A. Alonso, and I. M. de Diego, "Bio-inspired agent-based architecture for fraud detection," in *Proceedings of the 2020 3rd International Conference on Information Management and Management Science*, 2020. doi: 10.1145/3416028.3416039 pp. 67–71.
- [20] J. Wang, Y. Hong, J. Wang, J. Xu, Y. Tang, Q.-L. Han, and J. Kurths, "Cooperative and competitive multi-agent systems: From optimization to games," *IEEE/CAA Journal of Automatica Sinica*, vol. 9, no. 5, pp. 763–783, 2022. doi: 10.1109/JAS.2022.105506
- [21] P. Siswahyudi, T. A. Kurniawan, and V. Sugiarto, "Agent-oriented methodologies comparison: A literature review," *Advanced Science Letters*, vol. 24, no. 11, pp. 8710–8716, 2018. doi: 10.1166/asl.2018.12331
- [22] F. Bergenti, G. Caire, S. Monica, and A. Poggi, "The first twenty years of agent-based software development with jade," *Autonomous Agents and Multi-Agent Systems*, vol. 34, no. 2, pp. 1–19, 2020. doi: 10.1007/s10458-020-09460-z
- [23] J. Kazil, D. Masad, and A. Crooks, "Utilizing python for agent-based modeling: the mesa framework," in *International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation*. Springer, 2020. doi: 10.1007/978-3-030-61255-9_30 pp. 308–317.
- [24] B. K. Kler, "Tourism and restoration," in *Philosophical issues in tourism*. Channel View Publications, 2009. doi: 10.21832/9781845410988 pp. 117–134.
- [25] A. Vinerean, "Motivators that intervene in the decision making process in tourism," *Expert journal of marketing*, vol. 2, no. 2, 2014.
- [26] J. Swarbrooke and S. Horner, *Consumer behavior in tourism*. Heinemann, Oxford, 2007.
- [27] S. Amaro and P. Duarte, "An integrative model of consumers' intentions to purchase travel online," *Tourism management*, vol. 46, pp. 64–79, 2015. doi: 10.1016/j.tourman.2014.06.006
- [28] R. P. Falcao, J. B. Ferreira, and M. Carrazedo Marques da Costa Filho, "The influence of ubiquitous connectivity, trust, personality and generational effects on mobile tourism purchases," *Information Technology & Tourism*, vol. 21, no. 4, pp. 483–514, 2019. doi: 10.1007/s40558-019-00154-1
- [29] A. A. Mahrour and S. S. Hassan, "Achieving superior customer experience: An investigation of multichannel choices in the travel and tourism industry of an emerging market," *Journal of Travel Research*, vol. 56, no. 8, pp. 1049–1064, 2017. doi: 10.1177/0047287516677166
- [30] L. De Silva, F. R. Meneguzzi, and B. Logan, "Bdi agent architectures: A survey," in *Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI), 2020, Japão.*, 2020. doi: 10.17863/CAM.53101
- [31] L. S. Melo, R. F. Sampaio, R. P. S. Leão, G. C. Barroso, and J. R. Bezerra, "Python-based multi-agent platform for application on power grids," *International transactions on electrical energy systems*, vol. 29, no. 6, p. e12012, 2019. doi: 10.1002/2050-7038.12012

An Automated Algorithm for Fruit Image Dataset Building

Horea-Bogdan Mureşan

Babeş-Bolyai University

Faculty of Mathematics and Computer Science

No. 1, Mihail Kogălniceanu Street, Cluj-Napoca, Romania

Email: horea.muresan@ubbcluj.ro

Abstract—This paper introduces a new algorithm that utilises images from the Fruits-360 dataset, superimposes them of various backgrounds and creates associated annotation files with the coordinates of the bounding boxes surrounding the fruits. The main challenge of this task was accounting for the variations in lighting and occlusion associated with outdoor locations. The utility and application of such an algorithm is to reduce the need to collect real world data for training, accelerating the speed at which new models are developed. Using 3000 images generated by this algorithm we train a single shot multibox detector (SSD) to study the feasibility of using generated data during training. We then test the trained model on 70 real world images of apples (65 images of apples on trees and 5 images of apples in bunches) and obtain a mean average precision of 0.750 and we compare our results with those obtained by other state of the art models.

I. INTRODUCTION

THE most frequently mentioned challenge across papers that study the application of artificial intelligence in agriculture [1], [2], [3] was data scarcity [4], [5]. As such, authors have to collect images from orchards/plantations or obtain images from the Internet however, both methods have downsides. Collecting images from an orchard/plantation requires visiting the location during a certain time frame when the fruits are ripe [6]. Data collection from the Internet via scraping cannot guarantee image quality. Reference [7] suggests that multiple visits to the same or to different orchards during different periods of time are required for an ideal dataset. Both methods also require manual annotations, which is time consuming and is susceptible to human error [8]. One way to address the data scarcity problem is the creation of a data generating algorithm that can produce images that simulate real world conditions. We aimed to study the feasibility of using such generated data for training an apple detector.

The Fruits-360 dataset [9], which provides 90483 images of 131 fruits and vegetables, was introduced in 2017. The images in this dataset contain one fruit or vegetable per image with a white background. This makes the set very good for training classifiers, however it cannot be easily used to create detectors capable of locating/counting the number of fruits in an image. In order to extend the usability of the dataset for such tasks

as well, we have created an algorithm that generates artificial training images by superimposing fruits taken from the Fruits-360 dataset [10] on various backgrounds containing tree leaves, branches and other fruit. Alongside these artificial images, the software generates one annotation file per image containing all the coordinates of the bounding boxes that surround the fruits as well as the fruit class. Using a training and validation dataset of apple images generated with our algorithm we trained a SSD [11] and tested it on 70 real world images, obtaining an average precision of 0.750.

II. RELATED WORK

In paper [8], the authors proposed a novel approach based on convolutional neural networks for tomato fruit counting rather than area calculation. They used a modified version of the Inception-ResNet [12] network for this task. The authors noted that in order to obtain good performance with a deep learning algorithm, a dataset that captures all the variance in the conditions under which the model is expected to operate. Such datasets with annotated images were not available and the authors observe that they are difficult and time consuming to build. One factor that contributes to this is the limited time frame in which fruits are in the desired development stage for image taking. Another factor is human error, to which manual labelling is susceptible to. Thus, the authors created their own synthetic images by filling an image with green and brown circles to simulate background and then red circles to simulate tomatoes. Afterwards, a Gaussian blur filter was applied on the images. Training the model on a dataset consisting of exclusively synthetic images and then testing it on 100 real world images produced a 91% accuracy in estimating the fruit count.

In [13] several models for fruit detection were reviewed, such as multilayered perceptrons, LedNet [4] and InceptionV3 [14]. One issue that was present in all analysed papers was data scarcity. The authors of papers [5], [6] state that building an annotated dataset is a costly process from the perspectives of both time and material resources. For state of the art performance such a dataset should contain images of fruits in multiple

lighting and weather conditions, at different distances and at different levels of occlusions, according to the intended practical application of the model. Kang and Chen, in [4], used a multi-scale pyramid and clustering classifier to assist data labelling. Steven Chen et al. in [15] utilised a custom crowd-sourcing platform for quick data labelling to address this issue. As seen in [16], using transfer learning and a MobileNetV2 [17] model the authors achieved a good compromise between accuracy and inference speed. Similarly, Raheelin Siddiqi shows in paper [18] the effectiveness of transfer learning on classification accuracy of fruit images.

To the extent of our knowledge, outside of [8], no other projects have attempted to create a data generating algorithm that creates images of fruits on various background as well as the associated annotations.

III. METHODOLOGY

A. Data Generation

The goal of this algorithm is to allow a user to create the entirety or the majority of their training dataset without the need of seeking an orchard, collecting images and manually annotating them. This would reduce dependence on obtaining access to an orchard/plantation and on manual labelling. Furthermore, such an algorithm would allow experiments to be executed even when real world data cannot be collected. The idea behind this algorithm was inspired by paper [8], in which the authors created a dataset of synthetic images (green background with red dots) in order to simulate images of tomato plants. Using this dataset alone for training and then testing on real world images of tomato plants, the model trained by the authors achieved 91% accuracy.

The process relies on images from the Fruits-360 dataset [10] and on real images that will be used as backgrounds for the generated images. The image output size, class and maximum number of fruits contained in each such image can be customised to fit the scope of the project for which they are used.

The main steps of the algorithm that creates the dataset are:

- Randomly select a background image:
 - A folder of RGB background images (JPEG/PNG) of any resolution must be specified.
- Resize the selected background to the specified output width and height. We will refer to this image as a canvas.
 - The output width and height can be customized according to the purpose of the trained model. Similarly, random stretch can be applied to the image. This is done by specifying two intervals of floating point numbers (one for width, the other for height). From these intervals, a random float is selected and the respective dimen-

sion is multiplied by it. This simulates images taken using cameras with different aspect ratios (4:3, 16:9, etc.)

- Select fruit image from given labels.
 - Once the canvas is selected and resized, the algorithm randomly chooses a fruit image from the Fruits-360 dataset from one of the classes specified by the user.
 - The fruit image is resized to a randomly generated size within a given interval.
- Create mask and crop image to be centered on the fruit.
 - The images from the Fruits-360 dataset have a simple white background however, some fruits with a shiny texture, such as red apples, reflected the ambient light and contain white spots.
 - To ensure that, when isolating the fruit pixels, we do not remove the aforementioned white spots we apply the following operations:
 - 1) Firstly we create a mask by converting the fruit image to grayscale, then applying a threshold function such that the white pixels are transformed to black and the non-white pixels become white.
 - 2) Secondly, the mask is copied and a flood fill algorithm is applied starting from the corners.
 - 3) The flood filled mask copy is then inverted and a bitwise or is applied between it and the initial mask.
 - 4) Finally, the mask is eroded with a 3×3 kernel to eliminate border pixels between the fruit and the white background (Fig. 1).
- Augment fruit image.
 - The operations used to augment the fruit image are 90, 180, 270 degree rotations, brightness and contrast alteration and partial cropping.
 - Partial cropping simulates fruit occlusion by removing rows or columns of fruit pixels from the image. Both the probability of applying this

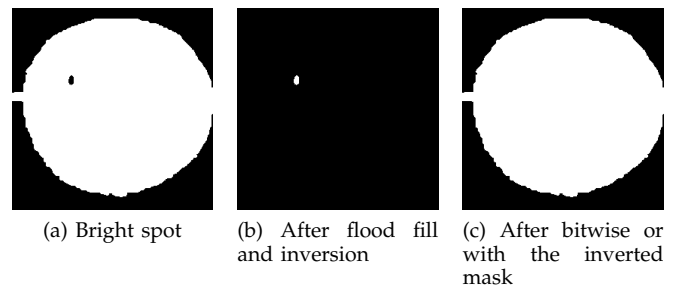


Fig. 1: Example of the bright spot issue caused by a granny smith apple and the solution to the problem.

TABLE I: PARAMETERS USED FOR GENERATING THE TRAINING AND VALIDATION DATA.

Max fruits per image	Min fruit size (px)	Max fruit size (px)	Training images	Validation images
50	30×30	250×250	500	100
30	150×150	400×400	1000	200
10	250×250	300×300	1000	200
25	50×50	500×500	500	100

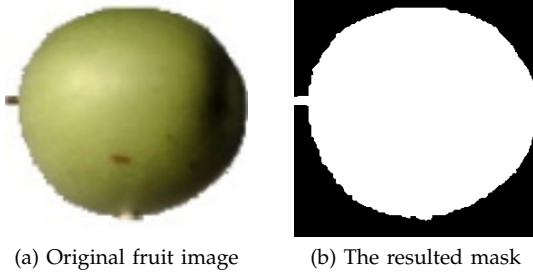


Fig. 2: An example of the mask-creating algorithm that selects only the fruit pixels while dropping the white background

operation and the maximum amount of pixel rows or columns that can be removed can be customized.

- Attempt to add the fruit image to the canvas.
 - The algorithm keeps a set of coordinates for each fruit added onto the canvas.
 - When a new fruit image can be added to the canvas, its coordinates are randomly generated. This is done to allow a more uniform distribution of fruits on a canvas, as well as to speed up the adding phase.
 - If a set of generated coordinates allows the image to be placed with an acceptable level of overlap, then the image is added onto the canvas. This is done by copying the fruit pixels from the fruit image with the help of the generated mask.
 - If, after a number of attempts, the image was not added, then the algorithm retries by resizing the image to 80% of its initial size, but not smaller than the user defined minimum size.
 - If the image is not successfully added even after resizing, then it is discarded and a new fruit image is selected.
- Once the fruits have been added, the canvas is saved in the PNG format with an index number as a name. The bounding boxes associated with the fruits in the image are stored in a separate file, either an xml or csv.
- The process is repeated until the requested number of images have been generated.

For the experiments done in this paper we generated a dataset of 3000 training images and 600 validation

images of size 768×1024 using all the apple classes from the Fruits-360 dataset and 30 background images. The background images were scraped from the Internet and contained foliage and trees. The images were generated according to the parameters presented in Table I. This distribution was chosen to simulate both a scenario in which fruits are close to the camera, which would produce an image with a few large fruits as well as the scenario in which a photo is taken from further away, in which case there would be numerous small fruits in the image.

B. Model Evaluation

In order to evaluate the feasibility of using a synthetic dataset to train a model that can then be used in real world scenarios, we selected the SSD512 model [11] as it has shown promising results in the area of intelligent agriculture, such as fruit detection [5] and leaf disease detection [19]. The SSD network was implemented in Keras [20] and was adapted for images of size 768×1024 . Training was done on the 3000 training images using an Adam optimizer with a learning rate of 0.0001 for the first 5 epochs and with a learning rate of 0.00005 for 5 more epochs. At the end of each epoch the model was evaluated on the 600 validation images and it was saved if it improved from the previous evaluation. The experiment was repeated 3 times, each time with a new set of generated images. The machine on which the training was done was equipped with an nVidia 2070 RTX GPU, 16 GB RAM and an Intel i7-8750H CPU. The implementation was done using TensorFlow 2.4 [21] and Python 3.7.7.

IV. RESULTS

For testing the model, a set of 70 real world images was created by scraping freely reusable images from the Internet and by taking photos of apples on trees or in bunches. The images contain multiple species of apples at medium to close distance, with some fruits being partially occluded by foliage or by other apples. The quality and resolution of the test images was varied, containing even some blurred apples, further increasing the difficulty of detecting them. The images were manually annotated using the **labelme** tool [22]. In order for an apple to be included in the annotations, at least half of it had to be visible in the image and had to be larger than 30×30 if the image is resized to 768×1024 .

Following, we will present the results of the trained SSD model on the 70 test images and compare them to

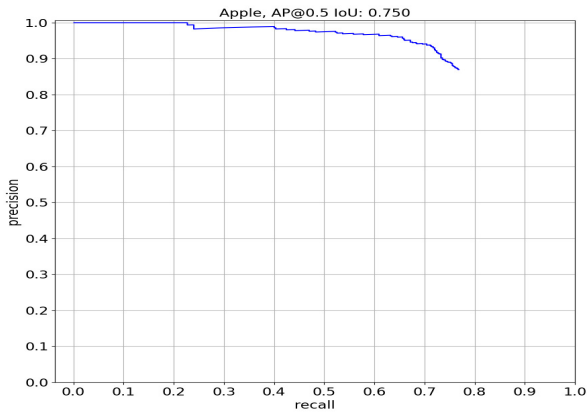


Fig. 3: Precision-Recall curve at 50% IoU.

existing fruit detectors. Table II shows the results of the three experiments conducted.

In Fig. 3 the precision-recall curve is presented for model number 2. It can be noted that the high precision value of this model means that the network produces very few false positives. However, the recall value does not exceed 80% for any of the three models. This indicates that the model fails to detect all apples from the test images, producing false negatives.

After visually inspecting the predictions on the test images, we identified several reasons behind the false negatives. Fig. 4 captures several examples:

- a large difference in lighting between a fruit and the others; as an example, Fig. 4d shows an apple that is underneath a cluster of leaves such that it does not receive the same amount of light as the other fruits in the image
- blurry images can impact the detection of fruits, as seen in Fig. 4a where a fruit located in the background is blurred compared to the apples in the foreground
- fruits that have a similar or identical colour with the background (green apples in this case) paired with partial occlusion produces situations where the fruits are not detected, shown in Fig. 4b and Fig. 4c

Table III presents the results of the models studied in paper [4], in which the authors tested two augmentation pipelines aiming to improve apple detection. The models from [4] were trained on real world images exclusively while our proposed model was trained on 3000 gener-

TABLE II: THE RESULTS OF THE THREE TRAINED MODELS.

Model	AP ₅₀	F1 Score	Recall	Precision
SSD#1	0.748	0.770	0.740	0.802
SSD#2	0.750	0.816	0.769	0.869
SSD#3	0.751	0.778	0.795	0.761

ated images. It can be noted that the performance of our proposed model is in the same vicinity, indicating the viability of training a model on generated data.

In paper [5] a Faster R-CNN and an SSD model were used for counting fruits from five classes: apple, orange, mandarin, lemon and tomato. Compared to our SSD model, the one proposed in [5] performs slightly better on fruit counting on the apple subset, achieving 81% accuracy, while ours achieved 76%. As mentioned previously, the model produces false negatives, explained by partially occluded fruits of the same color as the background or by blurred fruits, which can explain the difference between the predicted count and the actual count.

V. CONCLUSION AND FUTURE WORK

In this paper we have studied the feasibility of using a dataset of generated images as training data for an object detector and then applying it on real world images. We introduced an algorithm that uses the fruit images from the Fruits-360 dataset and custom background images to create new images alongside annotation files. We trained a SSD on an apple dataset of 3000 images generated with this algorithm and tested it using 70 real world apple images. Then we compared our results with other state of the art works and concluded that using generated images for training produces a model capable of handling real world data.

In the future, we plan to study if using generated data for training and using real world images for fine tuning leads to improved detection performance. Another development direction is the addition of more augmentation operations to the data generating algorithm to account for more data variance. A better way to simulate fruit occlusion, for example overlapping images of potential obstructions (eg. leaves, branches) over the fruit image could improve a model's capacity of detecting such fruits.

Overall, the presented study shows potential in using generated images instead of images collected from orchards or plantations as the principal source of training data for a fruit detector. We think that this paper serves as a starting point for other, more complex approaches.

REFERENCES

- [1] N. C. Eli-Chukwu, "Applications of artificial intelligence in agriculture: A review," *Engineering, Technology & Applied Science Research*, vol. 9, no. 4, pp. 4377–4383, 2019. doi: 10.48084/etasr.2756
- [2] O. Apolo-Apolo, J. Martínez-Guanter, G. Egea, P. Raja, and M. Pérez-Ruiz, "Deep learning techniques for estimation of the yield and size of citrus fruits using a uav," *European Journal of Agronomy*, vol. 115, p. 126030, 2020. doi: 10.1016/j.eja.2020.126030
- [3] S. Bargoti and J. P. Underwood, "Image segmentation for fruit detection and yield estimation in apple orchards," *Journal of Field Robotics*, vol. 34, no. 6, pp. 1039–1060, 2017. doi: 10.48550/arXiv.1610.08120

TABLE III: THE RESULTS OF OUR PROPOSED METHOD COMPARED WITH THOSE OBTAINED IN [4].

Method	AP ₅₀	F1 Score	Recall	Precision
LedNet (LW-Net)	0.826	0.834	0.821	0.853
LedNet (ResNet-101)	0.843	0.849	0.841	0.864
YOLOv3	0.803	0.803	0.801	0.82
YOLOv3 (Tiny)	0.782	0.783	0.776	0.796
Faster-RCNN (VGG)	0.814	0.818	0.814	0.835
Proposed method (SSD#2)	0.750	0.816	0.769	0.869

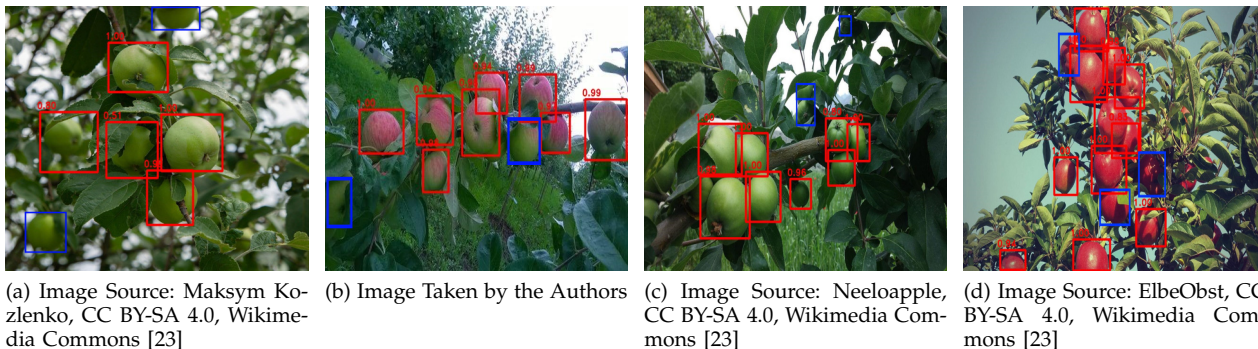


Fig. 4: Cases when the model did not correctly detect all apples. Shown with red are the model's predictions, while blue denotes missed detections.

- [4] H. Kang and C. Chen, "Fast implementation of real-time fruit detection in apple orchards using deep learning," *Computers and Electronics in Agriculture*, vol. 168, p. 105108, 2020. doi: 10.1016/j.compag.2019.105108. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0168169919314395>
- [5] H. Yu, S. Song, S. Ma, and R. O. Sinnott, "Estimating fruit crop yield through deep learning," in *Proceedings of the 6th IEEE/ACM International Conference on Big Data Computing, Applications and Technologies*, 2019. doi: 10.1145/3365109.3368766 pp. 145–148.
- [6] R. Kestur, A. Meduri, and O. Narasipura, "Mangonet: A deep semantic segmentation architecture for a method to detect and count mangoes in an open orchard," *Engineering Applications of Artificial Intelligence*, vol. 77, pp. 59 – 69, 2019. doi: 10.1016/j.engappai.2018.09.011. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0952197618301970>
- [7] A. Koirala, K. B. Walsh, Z. Wang, and C. McCarthy, "Deep learning—method overview and review of use for fruit detection and yield estimation," *Computers and Electronics in Agriculture*, vol. 162, pp. 219–234, 2019. doi: 10.1016/j.compag.2019.04.017
- [8] M. Rahnemoonfar and C. Sheppard, "Real-time yield estimation based on deep learning," in *Autonomous Air and Ground Sensing Systems for Agricultural Optimization and Phenotyping II*, J. A. Thomasson, M. McKee, and R. J. Moorhead, Eds., vol. 10218, International Society for Optics and Photonics. SPIE, 2017. doi: 10.1117/12.2263097 pp. 59 – 65.
- [9] H. Mureșan and M. Oltean, "Fruit recognition from images using deep learning," *Acta Universitatis Sapientiae, Informatica*, vol. 10, no. 1, pp. 26 – 42, 2018. doi: 10.48550/arXiv.1712.00580. [Online]. Available: <https://content.sciendo.com/view/journals/ausi/10/1/article-p26.xml>
- [10] M. Oltean and H. Muresan, "Fruits 360 dataset on github," 2017, [Online; accessed 16.09.2021]. [Online]. Available: <https://github.com/Horea94/Fruit-Images-Dataset>
- [11] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. E. Reed, C. Fu, and A. C. Berg, "SSD: single shot multibox detector," *CoRR*, vol. abs/1512.02325, 2015. doi: 10.1007/978-3-319-46448-0_2. [Online]. Available: <http://arxiv.org/abs/1512.02325>
- [12] C. Szegedy, S. Ioffe, and V. Vanhoucke, "Inception-v4, inception-resnet and the impact of residual connections on learning," *CoRR*, vol. abs/1602.07261, 2016. doi: 10.48550/arXiv.1602.07261. [Online]. Available: <http://arxiv.org/abs/1602.07261>
- [13] H. Mureșan, A. Călin, and A. Coroiu, "Overview of recent deep learning methods applied in fruit counting for yield estimation," *Studia Universitatis Babeș-Bolyai Informatica*, vol. 65, no. 2, pp. 50–65, 2020. doi: 10.24193/subbi.2020.2.04. [Online]. Available: <http://www.cs.ubbcluj.ro/~studia-i/journal/journal/article/view/58>
- [14] J. Fourie, J. Hsaio, and A. Werner, "Crop yield estimation using deep learning," in *7th Asian-Australasian Conference on Precision Agriculture*, 2017. doi: 10.5281/zenodo.893710 pp. 1–10.
- [15] S. W. Chen, S. S. Shivakumar, S. Dcunha, J. Das, E. Okon, C. Qu, C. J. Taylor, and V. Kumar, "Counting apples and oranges with deep learning: A data-driven approach," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 781–788, 2017. doi: 10.1109/LRA.2017.2651944
- [16] Q. Xiang, X. Wang, R. Li, G. Zhang, J. Lai, and Q. Hu, "Fruit image classification based on mobilenetv2 with transfer learning technique," in *Proceedings of the 3rd International Conference on Computer Science and Application Engineering*, 2019. doi: 10.1145/3331453.3361658 pp. 1–7.
- [17] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018. doi: 10.48550/arXiv.1801.04381 pp. 4510–4520.
- [18] R. Siddiqi, "Effectiveness of transfer learning and fine tuning in automated fruit image classification," in *Proceedings of the 2019 3rd International Conference on Deep Learning Technologies*, 2019. doi: 10.1145/3342999.3343002 pp. 91–100.
- [19] P. Jiang, Y. Chen, B. Liu, D. He, and C. Liang, "Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks," *IEEE Access*, vol. 7, pp. 59 069–59 080, 2019. doi: 10.1109/ACCESS.2019.2914929
- [20] P. Ferrari, "Ssd keras implementation," 2018, [Online; accessed 16.09.2021]. [Online]. Available: https://github.com/pierluigiferrari/ssd_keras
- [21] TensorFlow, "Tensorflow," 2015, [Online; accessed 16.09.2021]. [Online]. Available: <https://www.tensorflow.org>
- [22] K. Wada, "labelme," 2010, [Online; accessed 16.09.2021]. [Online]. Available: <https://github.com/wkentaro/labelme>
- [23] WikimediaCommons, "Wikimedia commons - freely usable media files," 2004, [Online; accessed 16.09.2021]. [Online]. Available: https://commons.wikimedia.org/wiki/Main_Page

NiaNet: A framework for constructing Autoencoder architectures using nature-inspired algorithms

Sašo Pavlič
 Faculty of Electrical Engineering
 and Computer Science
 University of Maribor
 Koroška cesta 46, 2000 Maribor
 Slovenia
 Email: saso.pavlic@student.um.si

Sašo Karakatič
 Faculty of Electrical Engineering
 and Computer Science
 University of Maribor
 Koroška cesta 46, 2000 Maribor
 Slovenia
 Email: saso.karakatic@um.si

Iztok Fister Jr.
 Faculty of Electrical Engineering
 and Computer Science
 University of Maribor
 Koroška cesta 46, 2000 Maribor
 Slovenia
 Email: iztok.fister1@um.si

Abstract—Autoencoder, an hourly glass-shaped deep neural network capable of learning data representation in a lower dimension, has performed well in various applications. However, developing a high-quality AE system for a specific task heavily relies on human expertise, limiting its widespread application. On the other hand, there has been a gradual increase in automated machine learning for developing deep learning systems without human intervention. However, there is a shortage of automatically designing particular deep neural networks such as AE. This study presents the NiaNet method and corresponding software framework for designing AE topology and hyper-parameter settings. Our findings show that it is possible to discover the optimal AE architecture for a specific dataset without the requirement for human expert assistance. The future potential of the proposed method is also discussed in this paper.

Index Terms—AutoML, autoencoder, deep learning, nature-inspired algorithms, optimization

I. INTRODUCTION

DEEP neural networks (DNN)s have seen a surge in popularity in recent years, with applications in various domains. Their potential began with better-than-human performance in tasks including image recognition, natural language processing, and product recommendation [1]–[3] and progressed to sophisticated tasks such as protein-folding, self-driving cars, and weather forecasting [4]–[6]. Even if DNN-based systems are not yet intelligent, we may employ them wisely to tackle complex problems. It is projected that with the current and future rise of computational resources and the large availability of data, DNN will benefit. Greater computer capabilities enable us to build more sophisticated and complex DNN topologies, while larger datasets will improve training performance.

Despite all the outstanding achievements and expectations of employing DNNs, data scientists and researchers are still dealing with DNN construction. The DNN construction phase can be a resourcefully expensive process and contributes to the global carbon footprint [7]. The construction phase specifies DNN topology, which includes defining layers, neurons, and connections. After the DNN topology has been designed,

the optimum hyper-parameters must be chosen. Variables such as topology and hyper-parameters influence the final performance of DNN, and those variables are often limited by the researcher’s prior knowledge and experience. The most significant disadvantage is the time spent by a human expert manually attempting to determine relevant variables for a specific search space.

In the literature, we can find many studies tackling the previously mentioned problem. The techniques presented in the studies attempt to automatically optimize certain aspects of the entire DNN creation process for a given search space [8]. One efficient method is to use a technique that employs population-based nature-inspired algorithms (NIA)s [9], [10]. Mostly because such a method is good at optimizing highly computationally expensive problems. When comparing the scale of studies, we can see that they are focusing on either the topology construction (without the weights) or topology with the weights simultaneously [11]. This is to keep the search space as small as possible. More details on the related work will follow in the following sections.

Motivated by these methods, we propose NiaNet (Nature-Inspired Algorithms for Deep Neural Network creaTion), a method capable of auto-designing the novel autoencoder (AE) model with only the dataset as input. The NiaNet is simultaneously constructing the AE topology and setting the optimal hyper-parameters. This process aims to optimally explore the search space by utilizing nature-inspired algorithms. The success of our proposed method, NiaNet, is determined by evaluating the best performing AE model with the fitness function, where reconstruction error, training duration, and topological simplicity are calculated. We find an advantage in our proposed method’s ease of use and adaptation to varied datasets while achieving promising results.

Altogether, the main contributions of this paper can be summarized as follows:

- We propose a method NiaNet for constructing the autoencoder topology and hyper-parameter setting.
- We present the automated machine learning (AutoML) framework capable of applying the NiaNet method on a given dataset.

This research was funded by the Slovenian Research Agency (research core funding No. P2-0057)

- We test the NiaNet method on a well-known diabetes dataset.

The remaining structure of this paper is as follows. Section II briefly describes the related work. Section III and Section IV presents the used framework and proposed method NiaNet. Obtained results of the experiment are presented in Section V. The last Section VI contains the conclusion.

II. RELATED WORK

In this section, we review relevant strategies for building DNN models without the need for human intervention. Since the construction of DNN is a highly complex problem that can also be represented as an optimization problem, scientists are looking for the potential of applying nature-inspired algorithms to cope with this problem. These algorithms are highly efficient in finding the solutions to multi-dimensional problems such as DNN construction. This section looks at target optimization problems that have received considerable attention in the literature. All of these strategies aim to discover the DNN topology efficiently in multiple dimensions. They vary in that they may either search simply the DNN topology (neurons and connections) or DNN topology and weights. The time axis or computational resources can represent efficiency before converging to the optimal solution.

A. Neuroevolution (NE)

A population of genetic encoding of artificial neural networks (ANN)s is evolved in neuroevolution to identify a network that solves a given task. Each encoding in the population (a genotype) is chosen one at a time and decoded into the neural network corresponding to it (a phenotype). The performance of this network in the task is then measured over time, yielding a fitness value for the relevant genotype. As a result, the process resembles an intelligent parallel search for superior genotypes, and it continues until a specific fitness threshold value is found or evolution reaches a specific generation limit [12]. Neuroevolution methods differ by type of encoding genotype to phenotype:

- A direct encoding will explicitly specify the direct connection between phenotype and genotype.
- An indirect encoding specifies the rules or parameters on how the phenotype is built from the genotype.

The following evolutionary algorithms, i.e., genetic algorithm (GA) [13], memetic algorithm (MA) [14], and particle swarm optimization (PSO) [15], showed promising results when tackling this problem.

Readers are invited to read a recent comprehensive study that presents current trends and future challenges in neuroevolution, as well as various types of neuroevolution and their strengths, and limitations [16].

B. Evolutionary neural architecture search

Neural Architecture Search (NAS) is a technique that automatically designs artificial neural networks. One of the many modifications of this technique is Evolutionary NAS (ENAS). It is a bio-inspired automated neural network architecture

design technique that follows the core principles of biological evolution [17]. Its goal is to identify a network topology that will give the best result on a given task. The three main components of the NAS method are as follows [18]:

a) Search space: It defines the boundaries inside which the search is allowed. This can be a set of rules for topology, layer number, layer type, and optimizers. Its size represents the set of all possibilities.

b) Search strategy: It defines the method for exploring the search space. The majority of the work on the NAS approach was focused on addressing this aspect. Since it is always challenging to determine which optimization methods work best and how to adapt or change them to yield better results. ENAS technique is using evolving ANN (EANN) as a search strategy. The EANN strategy is used to evolve ANN's connection weights, topology, and learning rules [19].

c) Evaluation strategy: Alternatively, sometimes called performance estimation strategy, evaluates the ANN offspring. Such evaluation is done prior to construction and training phase. This method primarily depends on many factors, such as search space size, datasets size, depth of topology, and others. To accurately measure the ANN offspring performance [18] many new methods have been proposed to reduce the time, and computation resources [20].

C. Structure learning

The method of utilizing data to train the linkages of a Bayesian network is known as structural learning. The method's goal is to represent the data in a graph format, providing a good balance of expressive power and querying performance. Bayesian networks are a type of structured knowledge representation in which domain variables are represented as nodes in a graph whose structure encodes their relationships [21]. However, these techniques need a lot of computing power, making the solution unsuitable for most applications with limited computing power and time.

III. AUTOMATED MACHINE LEARNING

As mentioned in the introduction, deep learning has been applied in various fields to solve challenging artificial intelligence (AI) tasks in recent years. Such diversification often leads to specific cases where even field experts operate on trial-and-error. This substantially increases the resources and time needed to create well-performing DNN models [22]. To reduce the development cost and automate the entire machine learning pipeline, an AutoML methodology was introduced. Its pipeline consists of data processing, feature engineering, model generation, and model evaluation. The goal is to be able to automate the complicated process of selecting pipeline components so that a user only needs to specify a dataset and an appropriate pipeline will be built automatically [23]. This frees up a human specialist to concentrate on other areas of the process.

This section introduces the AutoML framework, which utilizes our proposed method, NiaNet, in a model generation stage. In symbiosis, both the framework and method construct

a deep autoencoder topology and their hyper-parameters to discover the best possible ML pipeline for an input dataset. The framework (see. Fig. 1) is built using a layer-style layout architecture with multiple components.

A. Data Ingestion

Collecting and importing data into the ML pipeline is known as data ingestion. Acquiring data can be a complex component and one of the most challenging tasks because we need to have solid business and data understanding abilities [24]. The ML pipeline components will be influenced by the dataset used. The user performs this operation before the pipeline begins.

B. Dataset processing

Data processing is the first stage in the AutoML pipeline. There are numerous approaches for processing data to be used to build models. Real-world data is commonly skewed; there is missing data, which is often noisy. As a result, processing the data is required to make it clean and processed so that it may be run through the ML algorithms. The yellow section in Fig. 1 illustrates the data processing process. As authors in paper [22] explained, it may be divided into three processes: data collecting, data cleaning, and data augmentation. Data collection is an important stage in creating a new dataset or expanding an old one. The data cleaning process filters noisy data so that subsequent model training is not affected. Data augmentation is critical for increasing model robustness and improving model performance. The three aspects will be discussed in further depth in the following subsections. This stage is not yet automated in our framework.

C. Feature engineering

Feature engineering is the next stage in our AutoML pipeline. It usually consists of feature extraction, feature selection, and feature construction. In our ML pipeline, only the process feature selection is utilized. This process builds a feature subset based on the original set by reducing irrelevant or redundant features. This simplifies the model, preventing over-fitting and boosting model performance [22]. There are many manuals or automated ways of selecting the optimal feature set for a given dataset [10].

D. Model generation

Model generation is divided into two components, search space and optimization method, as shown in the 3rd section in Fig. 1. Where search space defines the AE topology and hyper-parameters. The AE architecture refers to a complete blueprint of DNN components such as:

- Topology shape (symmetrical, asymmetrical)
- Size of input, hidden and output layers
- Number of hidden layers
- Number of neurons in hidden layers

On the other side, in our AutoML framework, the following hyper-parameters are available in the search space:

- Activation function
- Number of epochs

- Learning rate
- Optimizer

Another component in the model generation stage is the optimization method. This component is responsible for finding the optimal solution within the edges of search space - parameter values to construct and train a given AE model. The solution is a one-dimensional array of elements from the above search space dimensions, each representing one of the parameters we are trying to optimize. The task of choosing the optimum solution is an iterative process. In this process, we are using the micro-framework NiaPy [25], which is an excellent tool for using the collection of nature-inspired algorithms for optimizing a given problem, such as ours. Each returned solution array from NiaPy framework is mapped according to equations [2-8]. More details are presented in section Proposed method. The AE model is created and trained in PyTorch using mapping rules that are controlled by the proposed method.

E. Model evaluation

The performance of an AE model must be evaluated after it has been constructed. The first process in the model evaluation phase is to use the DNN training and testing technique to evaluate each solution produced by the proposed method. However, this process requires a significant amount of time and computation resources. Once the AE model has been trained, the reconstruction loss and model complexity are evaluated based on the equations 10 and 11. In addition, the fitness function 11 is calculated, and the fitness value is passed back to the optimizer algorithm, which generates a new solution. Sections 3 and 4 of Fig. 1 show the communication flow between model generation and model evaluation.

F. Fittest AE model

After our iterative AutoML pipeline is completed, the fittest AE model is returned. To put it simply, the proposed model is optimal in terms of reconstruction loss and model complexity.

IV. PROPOSED METHOD

In the following section, we present in detail our proposed method, NiaNet ¹. This research will study whether an AE neural network with topology and hyper-parameters set by our proposed method will provide better encoding performance than AE designed manually. We believe that a nature-inspired search can discover a novel solution that may be hidden by human experts who are limited by their previous experience and knowledge. Our proposed method leans to be very straightforward and utilized on different datasets. This allows users to automatically perform some ML steps when using our AutoML framework, without manually searching for the right AE building blocks.

The method is based on applying the nature-inspired optimizer (we use a collection of algorithms in the NiaPy framework) to the problem of constructing AE typed neural networks. The solution is a one-dimensional array of seven

¹<https://github.com/SasoPavlic/NiaNet>

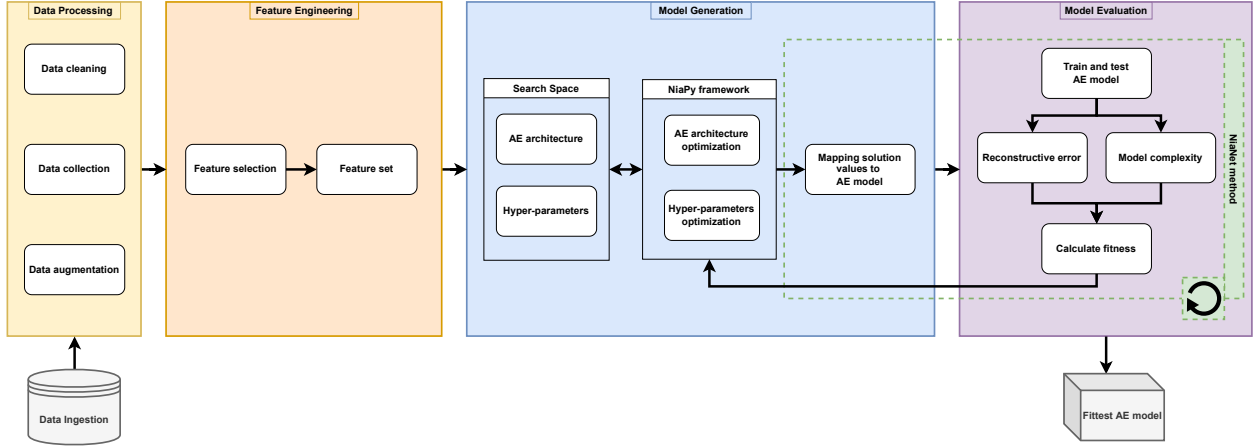


Fig. 1: A high-level overview of our AutoML pipeline, including data preparation (Section 1), feature engineering (Section 2), model generation (Section 3), and model evaluation (Section 4).

elements; each one represents one of the parameters we are attempting to optimize. The first three represent AE topology, and the last four represent hyper-parameters. The solution produced by the optimizer method is then mapped into the AE model using those representations. After the AE model has been built, it is evaluated using a fitness function that measures its performance. The fitness value represents the quality of the discovered solution. The fitness value is then reported back to the optimizer algorithm in the last step, allowing the search for the best optimal solution to continue. The algorithm is presented in Alg. 1.

Algorithm 1 Proposed method

Input: Dataset, parameters for NiaNet and NiaPy

Output: The fittest AE model

```

1: NiaNet.init()
2: NiaPy.init()
3: while terminationConditionNotMet do
4:   solution ← NiaPy.getBestSolution()
5:   shape ← NiaNet.mapShape(solution[0])
6:   layerStep ← NiaNet.mapLayerStep(solution[1])
7:   layers ← NiaNet.mapLayers(solution[2])
8:   activation ← NiaNet.mapActivation(solution[3])
9:   epochs ← NiaNet.mapEpochs(solution[4])
10:  LR ← NiaNet.mapLearningRate(solution[5])
11:  optimizer ← NiaNet.mapOptimizer(solution[6])
12:  fitness ← NiaNet.ModelEvaluation()
13:  NiaPy.generateNewSolution(fitness)
14: end while
15: fittestModel ← NiaNet.model(NiaPy.getBestSolution())
16: return fittestModel

```

A. Representation of individuals

Individuals in NiaNet are presented as real-valued vectors:

$$\chi_i^{(j)} = \{x_{i,0}^{(j)}, \dots, x_{i,n}^{(j)}\}, \text{ for } i = 0, \dots, \text{Np} - 1 \quad (1)$$

where each element of the solution is in the interval $\chi_{i,1}^{(j)} \in [0, 1]$. Real values in interval are then mapped according to equations [2-8], where y_1 stands for topology shape, y_2 for number of neurons per layer, y_3 for number of layers, y_4 for activation function, y_5 for number of epochs, y_6 for learning rate, y_7 for optimizer algorithm.

$$y_1 = \lfloor x[i] \rfloor; y_1 \in [0, 1] \quad (2)$$

$$y_2 = \lfloor \frac{x[i]}{\text{features}} \rfloor; y_2 \in [0, \text{features}] \quad (3)$$

$$y_3 = \lfloor \frac{x[i]}{\text{maxLayers}} \rfloor; y_{33} \in [0, \text{maxLayers}] \quad (4)$$

$$y_4 = \lfloor x[i] \rfloor; y_4 \in [0, 1] \quad (5)$$

$$y_5 = \lfloor x[i] * 10 + 100 \rfloor; y_5 \in [100, 200] \quad (6)$$

$$y_6 = \lfloor \frac{x[i]}{1000} \rfloor; y_6 \in [10^{-3}, 10^{-0}] \quad (7)$$

$$y_7 = \lfloor x[i] \rfloor; y_7 \in [0, 1] \quad (8)$$

The solution array is separated into two groups of indices, as shown in Fig. 2. The first three indices are used for topology mapping, while the fourth is utilized for hyper-parameter mapping. Together they form a complete solution that is subsequently used to build an AE model, as demonstrated in

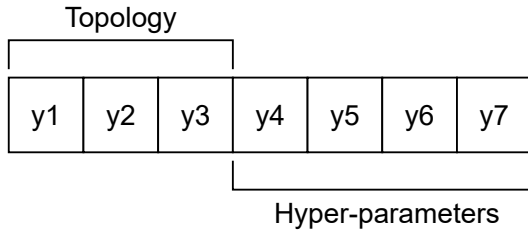


Fig. 2: An illustration of how indices in a solution array are allocated to variables $[y_1-y_7]$.

the algorithm 1. Each of these solutions is retrieved from the NiaPy library during the iterations of the NiaNet algorithm, with the goal of finding the most optimal solution. The fitness function determines the most optimal solution on a given dataset.

Topology representation: As we can see in Fig. 2 the first three elements in solution array are used for constructing the AE topology based on an element value. The first element y_1 is used to determinate the topology shape, which can be symmetrical or asymmetrical. This defines the shape relationship between the encoder and decoder parts. The number of neurons per layer is calculated using the second element y_2 . This is dependent on the number of features in dataset. Once we have the number of neurons y_2 we can calculate the number of layers y_3 in AE model. In the case of an asymmetrical AE model, the element value y_3 is used to set a random number of encoder layers before setting the remaining decoder layer number.

Hyper-parameters representation: Second part of solution array as seen in Fig. 2 is used for determining the hyper-parameters values which are utilized throughout the model training. The fourth y_4 and seventh y_7 elements are used for determining the activation function and optimizer algorithm based on a list of possible values. The fifth y_5 and sixth y_6 elements are used for number of epochs and learning rate depending on a defined range.

B. Encoding strategy

Once the real-valued solution array is proposed by the NiaPy framework, the real values are then mapped into the AE model according to the equations [2-8]. The element mapping value of solution array is determined with the binning process for all variables $[y_1-y_7]$. The bins are created in interval $\in [0, 1]$. Where each bin represents the possible mapping value. For example when mapping the AE's shape y_1 , the encoding algorithm takes the element's real value from solution array and map it to corresponding bin (symmetrical or asymmetrical). This can be seen in equation 9.

$$y_1 = \begin{cases} \text{symmetrical} & \text{if } x[i] \leq 0.5 \\ \text{asymmetrical} & \text{otherwise} \end{cases} \quad (9)$$

C. Fitness function

We defined the fitness function, which measures the individual solution by calculating its reconstruction error and

DNN complexity using equations 10 and 11. This enables us to analyze the encoding effectiveness and complexity of its topology.

$$E = \left(\sum_{i=1}^D (x_i - \hat{x}_i)^2 * 1000 \right) \quad (10)$$

$$C = \frac{(y_5)^2 + (y_3 * 100) + (\text{bottleneck_dim} * 10)}{100} \quad (11)$$

$$f(\chi_i^{(j)}) = \min E + C \quad (12)$$

Where E represents the reconstruction error, C topology complexity and $f(\chi_i^{(j)})$ represents the fitness value of an individual in evolution. Since equation 12 is designed to seek the global minimum, the fittest individual will be the one with the lowest fitness value. Furthermore, with the variable C , we address the issue of over-fitting. Less complex models are less likely to over-fit during the training process [26].

D. Training conditions

Due to research limitations, some parameters of the constructed AE model were static during the training phase. One of them is batch size, which is always set to 1, and another is the activation function, which remains the same once selected, across all encoder and decoder layers.

E. Data conditions

The data structure that the NiaNet method can process is limited to tabular data with only numerical values. As a result, any other type of data must be transformed first. We plan to expand our research in the near future to include time-series data as well.

V. EXPERIMENTS AND RESULTS

A. Introduction to dataset

In our experiments, we used a dataset that includes physiological data about the patients which are identified to have diabetes. The dataset is publicly available for everyone [27]. For each of the 442 diabetic patients, ten baseline characteristics, including age, gender, BMI, average blood pressure, and six blood serum measures, as well as the response of interest, a quantitative measure of disease progression one year after baseline, were collected. The eleventh feature is measuring the diabetes level. The dataset's feature values are represented solely by numerical values, with no missing values and a weak correlation (mean is 0.2). We standardized the data for each feature before using it, so that the distribution has a mean of 0 and a standard deviation of 1. This enables the DNN to be generated later with weights that are more similar across the features, resulting in more uniform topologies. Furthermore, the dataset was divided into the training and testing subset in a ratio 3:1.

B. Enviromental setup

The environment must be properly set up before running our AutoML framework with proposed method NiaNet [28]. In this section, we list the software components and configuration parameters that are utilized in a configuration file. All experiments were carried out using the Python programming language together with the libraries: NiaPy [25] for nature-inspired algorithms, Scikit-learn [29] for evaluating DNN models with metrics, NumPy [30] for working with arrays, PyTorch for DNN initialization [31]. Following computational resources were used for development and training environment: Razer Blade 15 Advanced (Early 2021 model - RZ09-036) with Intel i7-10875H CPU, Nvidia GeForce RTX 3080 with 8 GB GDDR6 memory and 6144 CUDA cores GPU, and 32 GB DDR4 RAM. Running on Windows 11 / Ubuntu 20.04.2 LTS.

C. Experimental settings

We utilized the values in Table I for NiaPy algorithm initialization settings. With parameters supplied, specified methods are utilized to search for optimal values in the encoded solution, which is expressed in equations [2-8]. The algorithms used in our experiment are: PSO [32], Differential Evolution (DE) [33], Firefly Algorithm (FA) [34], Self-adaptive Differential Evolution (jDE) [35], GA [36].

TABLE I: Used parameter values for NiaPy algorithms.

Parameter	Value
Dimensionality problem	7
Population size	default
Max evaluations	100
Runs	2
Lower bound	0.0
Upper bound	1.0

The following is an explanation for selected parameter values in table I: *Dimensionality problem* is set to 7 based on the solution array length, *Population size* is set to default, allowing NiaPy algorithms to have their own default value, *Max evaluations* is specifying number of evaluations on each algorithm separately, *Runs* is specifying number of repetitions (low number due to limited computational resources), *Lower bound* and *Upper bound* are borders within the real value number that can be represented in solution array.

Next we set the list of available activation functions and optimizers, which can be selected based on the mapping rules of y_4 and y_7 variables. Table II shows the activation functions, whereas table III shows the optimizers used in our experiment. All of the listed activation functions and optimizers, are available in PyTorch library, therefore any other ones that are not in the list can be easily added or removed.

TABLE II: List of activation functions in NiaNet method.

Activation function name
ELU
RELU
Leaky RELU
RRELU
SELU
CELU
GELU
Tanh

TABLE III: List of optimizers in NiaNet method.

Optimizer name
Adam
Adagrad
SGD
RAdam
ASGD
Rprop

D. Fittest Autoencoder architecture

We present the fittest AE model in this section, which was built and trained using the proposed NiaNet method in the AutoML framework. The experiment was carried out with the diabetes dataset with the above experimental parameter values. Produced solutions by the NiaNet method were evaluated by the fitness function in equation 12, where PSO produced the fittest solution.

Topology: The proposed solution array was mapped into the AE model based on the encoding strategy. The proposed model was a symmetrical AE, having a single-layer encoder and decoder. The encoder was built to take a 10-D input vector and compress it into an 8-D latent vector. The decoder was a mirrored encoder that took an 8-D latent vector as input and decompressed it back to a 10-D output vector. This indicates that the initial 10-D vector was compressed by the 20%. Another good indication can be seen in a simplicity of a model in terms of AE deepness (see. Fig. 3).

Hyper-parameters: When mapping up the elements in solution, we got the hyper-parameters for the previously mentioned topology. Where an activation function = *RRELU*, number of epochs = 110, learning rate = 0.11 and optimization algorithm = *RAdam*.

Achieved performance: On the testing dataset, we applied the root mean squared error (RMSE) to objectively quantify the fittest AE model's performance. It allows us to examine how close the reconstructed data examples (output) are from the ground truth (input) on average. The closest the result of RMSE metric is to zero, the smaller the difference between input and output. In our case, the fittest AE model reached the value of 0.11, which can be the starting point for future research.

E. Results by optimization algorithm

The following are the findings of our more in-depth analysis. Each of the fittest solutions produced by algorithms PSO, DE, FA, jDE, and GA is listed in table IV. The PSO algorithm

Algorithm	Fitness value	RMSE	Bottleneck size	Topology shape	number of neurons per layer	Layers in AE	Activation function	Epochs	Learning rate	Optimizer algorithm
PSO	231	0.11	8	symmetrical	En[8], De[10]	2	RRELU	110	0.11	RAdam
FA	353	0.15	8	symmetrical	En[8], De[10]	2	SELU	140	0.26	Adagrad
jDE	523	0.40	5	symmetrical	En[5], De[10]	2	RRELU	110	0.38	Adagrad
GA	553	0.40	9	symmetrical	En[9], De[10]	2	TANH	120	0.04	ASGD
DE	556	0.26	9	symmetrical	En[9], De[10]	2	CELU	170	0.06	RAdam

TABLE IV: The NiaNet fittest solutions found by selected algorithms

achieved a significantly higher fitness value, closely followed by the FA algorithm. Whereas all other algorithms ended up with very comparable fitness values, despite arriving at different solutions in AE model building blocks. This can be explained by looking at the formulation of our fitness function, equation 12. When looking at the proposed bottleneck sizes, we see the values span from 5 to 9. Since the input shape is 10 for the selected dataset, this is relatively considerable variability between algorithms. While the variables such as topology shape, epochs, and number of layers are nearly identical. All the algorithms found the optimal solution in those variables for this problem. The number of neurons in each layer varies between algorithms, since it is related to the size of the bottleneck. Multiple solutions for activation function, learning rate and optimizers were also proposed.

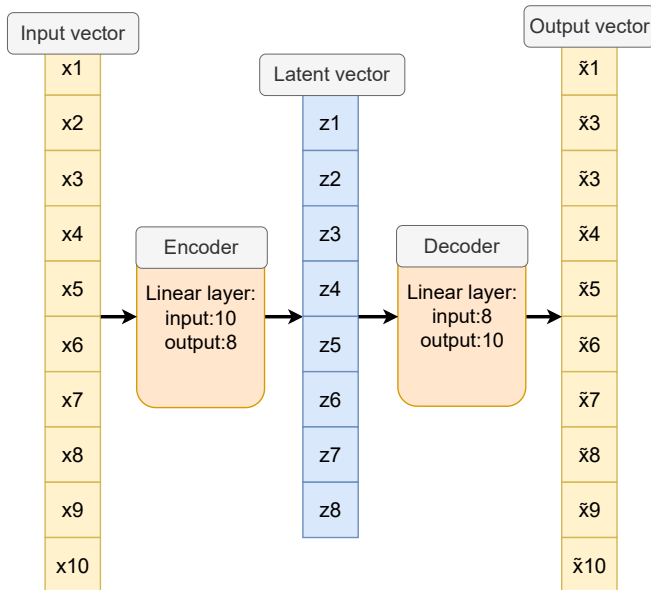


Fig. 3: Fittest autoencoder topology, designed by PSO algorithm.

VI. CONCLUSION

This paper presented NiaNet, a novel method for building AE models based on the AutoML methodology. The method sets up the topology and the hyper-parameters based on the solutions designed by nature-inspired algorithms. The results gathered in experiments are indicating a promising avenue that has to be further explored. This could help reduce the human resources needed in the model generation stage of AutoML.

Based on these exciting findings, we plan to expand our research toward finding an optimal solution for a broader

range of training parameters and AE topologies with various depth, width, and layer types. The objective for the future is to compare our proposed method to existing AutoML methodologies in a more extensive performance comparison using a variety of datasets. Having numerous solutions for various datasets could provide us with insights into how to build optimal AE models in the future.

REFERENCES

- [1] F. Yu, Z. Qin, C. Liu, D. Wang, and X. Chen, "REIN the RobuTS: Robust DNN-Based Image Recognition in Autonomous Driving Systems," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 40, no. 6, pp. 1258–1271, Jun. 2021, conference Name: IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems.
- [2] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, L. Kaiser, S. Gouws, Y. Kato, T. Kudo, H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. Smith, J. Riesa, A. Rudnick, O. Vinyals, G. Corrado, M. Hughes, and J. Dean, "Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation," *arXiv:1609.08144 [cs]*, Oct. 2016, arXiv: 1609.08144. [Online]. Available: <http://arxiv.org/abs/1609.08144>
- [3] S. Shekhar, A. Singh, and A. K. Gupta, "A Deep Neural Network (DNN) Approach for Recommendation Systems," in *Advances in Computational Intelligence and Communication Technology*, ser. Lecture Notes in Networks and Systems, X.-Z. Gao, S. Tiwari, M. C. Trivedi, P. K. Singh, and K. K. Mishra, Eds. Singapore: Springer, 2022, pp. 385–396.
- [4] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Židek, A. Potapenko, A. Bridgland, C. Meyer, S. A. A. Kohl, A. J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, E. Clancy, M. Zielinski, M. Steinegger, M. Pacholska, T. Berghammer, S. Bodenstein, D. Silver, O. Vinyals, A. W. Senior, K. Kavukcuoglu, P. Kohli, and D. Hassabis, "Highly accurate protein structure prediction with AlphaFold," *Nature*, vol. 596, no. 7873, pp. 583–589, Aug. 2021, number: 7873 Publisher: Nature Publishing Group. [Online]. Available: <https://www.nature.com/articles/s41586-021-03819-2>
- [5] Z. Li, M. Pan, T. Zhang, and X. Li, "Testing DNN-based Autonomous Driving Systems under Critical Environmental Conditions," in *Proceedings of the 38th International Conference on Machine Learning*. PMLR, Jul. 2021, pp. 6471–6482, iSSN: 2640-3498. [Online]. Available: <https://proceedings.mlr.press/v139/li21r.html>
- [6] J. N. K. Liu, Y. Hu, Y. He, P. W. Chan, and L. Lai, "Deep Neural Network Modeling for Big Data Weather Forecasting," in *Information Granularity, Big Data, and Computational Intelligence*, ser. Studies in Big Data, W. Pedrycz and S.-M. Chen, Eds. Cham: Springer International Publishing, 2015, pp. 389–408. [Online]. Available: https://doi.org/10.1007/978-3-319-08254-7_19
- [7] P. Dhar, "The carbon impact of artificial intelligence," *Nature Machine Intelligence*, vol. 2, no. 8, pp. 423–425, 2020.
- [8] E.-G. Talbi, "Automated Design of Deep Neural Networks: A Survey and Unified Taxonomy," *ACM Computing Surveys*, vol. 54, no. 2, pp. 34:1–34:37, Mar. 2021. [Online]. Available: <https://doi.org/10.1145/3439730>
- [9] G. Vrbančić, I. Fister jr, and V. Podgorelec, *Designing Deep Neural Network Topologies with Population-Based Metaheuristics*, Sep. 2018.
- [10] L. Pečnik and I. Fister, "NiaAML: AutoML framework based on stochastic population-based nature-inspired algorithms," *Journal of Open Source Software*, vol. 6, no. 61, p. 2949, May 2021. [Online]. Available: <https://joss.theoj.org/papers/10.21105/joss.02949>

- [11] V. K. Ojha, A. Abraham, and V. Snášel, "Metaheuristic design of feedforward neural networks: A review of two decades of research," *Engineering Applications of Artificial Intelligence*, vol. 60, pp. 97–116, Apr. 2017. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0952197617300234>
- [12] R. Miikkulainen, "Neuroevolution."
- [13] K. O. Stanley and R. Miikkulainen, "Evolving neural networks through augmenting topologies," vol. 10, no. 2, pp. 99–127. [Online]. Available: <https://direct.mit.edu/evco/article/10/2/99-127/1123>
- [14] A. Conradie, R. Miikkulainen, and C. Aldrich, "Intelligent process control utilising symbiotic memetic neuro-evolution," in *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No.02TH8600)*, vol. 1, pp. 623–628 vol.1.
- [15] A. Hara, J.-i. Kushida, K. Kitao, and T. Takahama, "Neuroevolution by particle swarm optimization with adaptive input selection for controlling platform-game agent," in *2013 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 2504–2509, ISSN: 1062-922X.
- [16] E. Galván and P. Mooney, "Neuroevolution in Deep Neural Networks: Current Trends and Future Challenges," *IEEE Transactions on Artificial Intelligence*, vol. 2, no. 6, pp. 476–493, Dec. 2021, conference Name: IEEE Transactions on Artificial Intelligence.
- [17] C. Broni-Bediako, "Automated Deep Neural Networks with Gene Expression Programming of Cellular Encoding - Towards the Applications in Remote Sensing Image Understanding-," Mar. 2022. [Online]. Available: https://soka.repo.nii.ac.jp/index.php?active_action=repository_view_main_item_detail&page_id=13&block_id=68&item_id=40743&item_no=1
- [18] T. Elsken, J. H. Metzen, and F. Hutter, "Neural Architecture Search: A Survey," *arXiv:1808.05377 [cs, stat]*, Apr. 2019, arXiv: 1808.05377. [Online]. Available: <http://arxiv.org/abs/1808.05377>
- [19] X. Yao, "Evolving artificial neural networks," *Proceedings of the IEEE*, vol. 87, no. 9, pp. 1423–1447, Sep. 1999, conference Name: Proceedings of the IEEE.
- [20] E. Thomas, M. Jan Hendrik, and H. Frank, "Neural Architecture Search: A Survey," *arXiv:1808.05377 [cs, stat]*, Apr. 2019, arXiv: 1808.05377. [Online]. Available: <http://arxiv.org/abs/1808.05377>
- [21] M. Scanagatta, A. Salmerón, and F. Stella, "A survey on Bayesian network structure learning from data," *Progress in Artificial Intelligence*, vol. 8, no. 4, pp. 425–439, Dec. 2019. [Online]. Available: <https://doi.org/10.1007/s13748-019-00194-y>
- [22] X. He, K. Zhao, and X. Chu, "AutoML: A survey of the state-of-the-art," *Knowledge-Based Systems*, vol. 212, p. 106622, Jan. 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0950705120307516>
- [23] B. Evans, "Population-based Ensemble Learning with Tree Structures for Classification," thesis, Open Access Te Herenga Waka-Victoria University of Wellington, Jan. 2019. [Online]. Available: https://openaccess.wgtn.ac.nz/articles/thesis/Population-based_Ensemble_Learning_with_Tree_Structures_for_Classification/17136296/1
- [24] J. Meehan, N. Tatbul, C. Aslantas, and S. Zdonik, "Data ingestion for the connected world," p. 11.
- [25] G. Vrbančič, L. Brezočnik, U. Mlakar, D. Fister, and I. Fister, "NiaPy: Python microframework for building nature-inspired algorithms," *Journal of Open Source Software*, vol. 3, no. 23, p. 613, Mar. 2018. [Online]. Available: <https://joss.theoj.org/papers/10.21105/joss.00613>
- [26] C. M. Bishop, *Neural Networks for Pattern Recognition*. USA: Oxford University Press, Inc., 1995, p. 332.
- [27] Diabetes data. [Online]. Available: <https://www4.stat.ncsu.edu/~boos/var.select/diabetes.html>
- [28] "NiaNet/autoencoder.py at 408b7fe0f4634439eb69e75f6b0c5afb18ce0702 · SasoPavlic/NiaNet." [Online]. Available: <https://github.com/SasoPavlic/NiaNet>
- [29] "scikit-learn: machine learning in Python — scikit-learn 1.0.2 documentation." [Online]. Available: <https://scikit-learn.org/stable/>
- [30] "NumPy." [Online]. Available: <https://numpy.org/>
- [31] PyTorch. [Online]. Available: <https://www.pytorch.org>
- [32] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-international conference on neural networks*, vol. 4. IEEE, 1995, pp. 1942–1948.
- [33] R. Storn and K. Price, "Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces," *Journal of global optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [34] X.-S. Yang, *Nature-inspired metaheuristic algorithms*. Luniver press, 2010.
- [35] J. Brest, S. Greiner, B. Boskovic, M. Mernik, and V. Zumer, "Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems," *IEEE transactions on evolutionary computation*, vol. 10, no. 6, pp. 646–657, 2006.
- [36] J. H. Holland, *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press, 1992.

Aspects of autonomous drive control using NVIDIA Jetson Nano microcomputer

Kacper Podbucki

Poznan University of Technology, ul. Jana Pawła II 24,
60-965 Poznań, Poland

Email: kacper.podbucki@put.poznan.pl

Tomasz Marciniak

Poznan University of Technology, ul. Jana Pawła II 24,
60-965 Poznań, Poland

Email: tomasz.marciniak@put.poznan.pl

□

Abstract— The article describes the training process and experiments regarding autonomous movement by the autonomous car Waveshare JetRacer AI. The central unit responsible for controlling the vehicle's systems, i.e. the steering servo and the DC motors used for the drive, is the NVIDIA Jetson Nano embedded device. The application of the IMX219 camera module for data acquisition and training of a neural network models on microcomputer and their use for the implementation of autonomous driving are described.

I. INTRODUCTION

THE autonomous movement of vehicles is a task that requires a careful analysis of the environment. For this purpose, vision or sensory systems are used. They play the role of a source of data about the environment in specific assistance systems, e.g., active cruise control, emergency braking, environment mapping, digital side and rear view mirrors, parking assist and lane assist [1]-[3]. Due to the advancement of such systems and their continuous development, it is possible that they will soon allow the achievement of higher levels of driving automation, ultimately leading to the mass production of fully autonomous vehicles.

There are six levels of vehicle driving autonomy defined by SAE International in the SAE J3016 standard, which are numbered from 0 to 5 [4]:

- Level 0: No Automation
- Level 1: Driver Assistance
- Level 2: Partial Automation
- Level 3: Conditional Automation
- Level 4: High Automation
- Level 5: Full Automation.

At level 0 there is no automatic vehicle control, but the system may issue warnings. In the next stage, it is defined that the driver must be ready to take control of the vehicle at any time. The automated system may be equipped with features such as Adaptive Cruise Control (ACC), Parking Assistance with automated steering, and Lane Keeping Assistance (LKA) Type II in any combination. The second level of partial automation obliges the driver to detect objects and events and to be ready for response if automated system fails to react properly. A computer executes accelerating, braking, and

steering, but can be immediately deactivated while driver takes over the control. Level 3 allows the driver to safely turn attention from driving tasks within known, limited environments (e.g. freeways). But if any warning or alert occurs, the driver must intervene. In the 4th level the vehicle is controlled by an automated system but only when the conditions in the environment allow the enabling system by the driver. When running, there is no need to pay attention to the driver while driving. The last level requires only to set the destination and press the start button. The automatic system will drive to any location where it is legal to drive [5].

Real-time processing of data received from vision modules or sensors and sending feedback control signals requires the use of computers with high computing power. Due to the dynamic development of embedded devices and increasing their efficiency, it is possible to use them for tasks that require real-time calculations [6]. Examples of such modules are microcomputers from the NVIDIA Jetson family. They are equipped with powerful GPUs that allow performing complex calculations using models of neural networks.

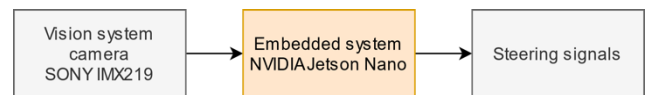


Fig. 1 Block diagram of the tested system



Fig. 2 Waveshare JetRacer AI with NVIDIA Jetson Nano microcomputer and Waveshare IMX219 camera

The purpose of processing the captured video sequences is to obtain parameters that allow precise control of the steering system of the vehicle (Fig. 1). The implementation of algorithms for autonomous driving in the model environment was carried out with the use of a four-wheeled Waveshare JetRacer AI mobile vehicle with a NVIDIA Jetson Nano

□ This research was funded partly by the 2022 subvention and partly with the SMART4ALL EU Horizon 2020 project, Grant Agreement No 872614.

minicomputer and a Waveshare IMX219 camera (Fig. 2). Various models of neural networks were trained and tested for the autonomous driving of a vehicle along a designated route. The software allows to stream the image from the car's on-board camera and enables the user to communicate with the environment by controlling the displayed widgets with the use of a mouse or an external controller.

II. WAVESHARE JETRACER AI HARDWARE BASED ON NVIDIA JETSON MICROCOMPUTER

Waveshare JetRacer AI is a platform that allows to independently construct a four-wheeled mobile vehicle. The kit consists of mechanical elements of the chassis, bumpers, camera arm, and wheels. The steering system is controlled by the MG996R servo with a torque of 9 kg/cm. The drive is provided by two 37-520 DC motors with a reduction rate of 1:10 and idle speed of 740 RPM. The 12.6V power supply is provided by the use of 3 Li-Ion 18650 batteries connected in series. Wireless communication, which allows remote programming and car control, is carried out using the AC8265 WiFi module with two antennas. The main PCB is equipped with a 0.91" OLED display with a resolution of 128×32 pixels. It shows the parameters for the use of computer resources and the IP address used to establish wireless communication. The manufacturer also includes a universal gamepad that can be used to control the car [7].

The central unit responsible for controlling the steering and drive systems as well as processing data from the camera is the NVIDIA Jetson Nano B01 microcomputer. Its most important technical specifications [8] are presented in Table I. This microcomputer was designed mainly for use in tasks related to artificial intelligence. It allows running many neural networks, which can perform processes such as image classification or object detection. Their simultaneous operation while maintaining appropriate performance is ensured by the CUDA architecture, which allows for the performance of complex and computationally expensive operations, such as matrix calculations or 3D rendering using the potential of the CPU and GPU [9].

TABLE I.
NVIDIA JETSON NANO B01 SPECIFICATION

GPU	128-core Maxwell
CPU	Quad-core ARM A57 1.43 GHz
RAM	4 GB 64-bit LPDDR4
Camera	2×MIPI CSI-2 DPHY lanes
Connectivity	M.2 Key e.g. for network card
Operating System	Linux for Tegra – JetCard 4.5.1

Image recording is possible by equipping the vehicle with a Waveshare IMX219 camera module 160 degree FoV [10]. Its resolution is 8 megapixels and the lens is wide-angle 160 degrees. This shows the significant extension of the perspective in comparison to 62.2 degrees in the commonly used Raspberry Pi Camera v2. The Waveshare IMX219 camera is also characterized by an aperture of f2.35 and a

focal length of 3.15mm. It allows to take pictures with a maximum resolution of 3280×2464 pixels and video recording with a resolution of 1080p and frequency of 30 frames per second. The camera module itself (without dedicated PCB) is compatible with both the Raspberry Pi Camera v2 PCB and the NVIDIA Jetson Nano minicomputer used by connecting via the CSI connector.

Waveshare JetRacer AI car programming can be performed through a wireless connection to the NVIDIA Jetson Nano microcomputer. For this purpose, the JupyterLab environment is used, which is installed by default in the software package provided by the manufacturer. The system also has implemented libraries necessary to support neural networks and video sequence processing operations, such as: OpenCV, Tensorflow, NumPy, PyTorch, or NVIDIA TensorRT [11].

III. DATA COLLECTION FOR NEURAL NETWORK TRAINING

The task for the Waveshare JetRacer AI vehicle is to follow an oval track, on which the edges are marked with yellow lines and the central axis with a white dotted line [12]. Based on the image from the camera, which is sent to the car's main computer, the neural network algorithm decides to correct the direction of the vehicle's movement by changing the position of the steering servo. In the presented study, movement at variable speed is not assumed. Its value is manually controlled by the program widget, but it is possible to extend the algorithm functionality of its autonomous selection in the future.

The implementation of the Waveshare JetRacer AI car autonomous movement algorithm consists of several stages. The first one is the hardware configuration of the device, i.e. the appropriate mechanical setting of the steering system and empirical determination of its correction coefficients and the parameters of the drive system. They depend on the design of a specific car. After establishing communication with the computer via the wireless network, it is possible to adjust them through the widgets included in the program. Then, in the next step, for convenience of use, it is worth connecting the wireless controller to the computer to be able to remotely control the vehicle. For this purpose, can be used the gamepad provided by the manufacturer or any other controller such as the Xbox 360 Controller.

In the next stage, the camera widget is initialized to show a preview on one screen and on the second one to precisely define the coordinates of the direction vector, according to which the car should move autonomously. These values can be set using the widget sliders, or after appropriate code modification with a wireless controller. The user interface and the actual vehicle position are shown in Figs. 3 and 4.

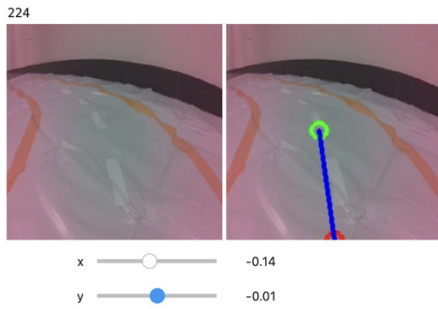


Fig. 3 User interface camera view in JupyterLab widget for training data collection



Fig. 4 Actual car position on track during data collection

Having prepared all the described tools and the vehicle on the track in the target environment, the neural network model training can be started. This process should begin with collecting training and validation data. It involves the car going over the track several times, but not continuously and recording the video sequence, but by stopping, changing the position, and taking pictures.

The key information, apart from the camera frame itself, is also the coordinates of the vector of the direction in which the car should move. They are saved in the file name along with a unique identifier for the image. Setting the vector position is a task that must be performed manually using the interface sliders and preview on the widget, or with the use of a controller.

It is required to prepare the dataset in such a way that it contains data allowing to choose the optimal path for the vehicle, but also boundary conditions, e.g. car on the edge of the track for the case to return to the correct trajectory. The larger the dataset that defines how a car behaves, the greater the chance of more precise autonomous driving.

IV. IMPLEMENTATION OF A NEURAL NETWORK FOR AUTONOMOUS VEHICLE MOVEMENT

The key element of the software that allows the implementation of the task of autonomous vehicle movement is the appropriate selection of the neural network model. The concept of the structure of the implemented solution is presented in Fig. 5.

The algorithm for autonomous driving of the Waveshare JetRacer AI car was based on the ResNet18 neural network. This architecture was chosen for its effectiveness at the level of other solutions such as VGG. The advantage of ResNet solutions, however, is the small size of the model at the level of 22.7 MB [13]. The architecture of this convolutional neural

network is 18 layers deep (17 convolutional and one connected) [14]. The shape of its last layer has been changed by the Linear function from the PyTorch library, and finally it has the same number of results as the number of classes in the dataset. The Linear PyTorch function creates a single-layer unidirectional feedforward network in which data only flows from input to output. An important aspect of this type of solution is backpropagation, which consists of improving at each step of the training process due to the correction of weights based on the estimation of the error made by neurons during training. The Adam algorithm was responsible for optimizing the network model with such a structure. The set of collected data (600 frames) was divided as follows: 80% of the training images and 20% of the validation data. To optimize the performance of the software, it was decided to convert the PyTorch model to the TensorRT model, which enables more than three times the increase in the number of processed frames per second, from 29.4 to 90.2 FPS [15].

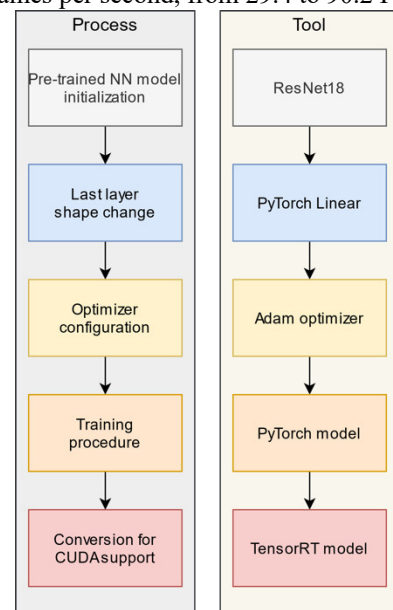


Fig. 5 Block diagram of neural network model for autonomous driving During the training of the neural network, an experiment

TABLE II.
MEAN LOSS FUNCTION VALUE FOR DIFFERENT EPOCHS NUMBER

Epochs number	Training data	Validation data	Training time [s]
30	0.042	0.035	754.7
50	0.024	0.018	1229.6
100	0.014	0.008	2459.1

was carried out to check the impact of the number of epochs on the loss function and the behavior of the vehicle in autonomous driving mode. The epoch is the one-time use of all the cases included in the training set during the whole teaching process. These repetitions are intended to perform many steps of the training algorithms until the error in the output is acceptably small. The following values of the number of epochs were checked during the experiments: 30,

50 and 100. The mean values of the loss function and training times are summarized in Table II.

The average values obtained for the loss function show the high efficiency of the neural network model training process. As expected, the dependence of the decrease in the function value with the increase in the number of epochs can be noticed. The results for the validation data are slightly lower than for the training images, but these are average values, so there is no risk of overfitting phenomenon. The analysis of the complete input data confirmed that the structure of the network, the ratio of the number of training and validation images, as well as the selection of network parameters made it possible to achieve results that allow testing under real conditions.

Due to the use of the neural network model, when the vehicle is moving, an image is taken from the camera, which, after analysis, returns the value of the function responsible for steering the torsion of the front axle of the platform, depending on the model's prediction. In addition, parameters such as steering gain and steering bias have also been implemented, which have to be set manually because they are the result of the physical imperfections of the vehicle's design and the servo used for the steering system.

Depending on the number of epochs implemented during the neural network training process, the following observations were made during real tests. The car, using a model trained for 30 epochs, tended to cross the orange lines at the exit of the curve and cut them off. This problem was regular and repetitive, since crossings of the track limits occurred more or less at the same points on the route. This phenomenon was eliminated for the model with 50 epochs. The vehicle was already moving inside the orange lines, in accordance with the planned trajectory, but on straight sections its path deviated from the central axis marked by a white dotted line. This effect was completely eliminated for a model trained with 100 epochs. The car was driven without problems along the set trajectory, not exceeding the limits of the track and keeping its position in the axis of the route.

V. CONCLUSIONS

The tested model of the Waveshare JetRacer AI vehicle allowed for the implementation of an algorithm for the task of autonomous driving along a designated route. Due to the use of the NVIDIA Jetson Nano B01 microcomputer supporting the GPU-based CUDA architecture, it was possible to run a complex model of the neural network. It processes the data obtained from the Waveshare IMX219 160 degree FoV camera, and then sends the control signals to the servo responsible for the steering system.

Preparation of training data for a dataset that reflects the test environment, selection of appropriate parameters of the neural network, and optimization of its performance allow for obtaining a reliable algorithm for autonomous driving. It should be noted that it does not require powerful computing power from efficient graphics cards dedicated to PC devices, but can also be successfully used on embedded devices [16].

The conducted experiments confirm the correct operation of both the hardware and the software part. After an appropriate selection of the number of epochs during the neural network model training process, it is possible to eliminate undesirable behavior of the vehicle, such as going beyond the boundaries of the route or deviation from the set movement trajectory. The trajectory of the movement of Waveshare JetRacer AI model based on the algorithm analyzed is stable.

Further research will allow the system to be expanded with additional sensor modules for more precise environment monitoring (including obstacles detection) using data fusion techniques [17].

REFERENCES

- [1] Yeong, D.J.; Velasco-Hernandez, G.; Barry, J.; Walsh, J. Sensor and Sensor Fusion Technology in Autonomous Vehicles: A Review. *Sensors* 2021, 21, 2140. <https://doi.org/10.3390/s21062140>
- [2] Baiou M., Quilliot A., Aduane L., Mombelli A., and Zhu Z., Algorithms for the Safe Management of Autonomous Vehicles, *Annals of Computer Science and Information Systems. IEEE*, Sep. 26, 2021. doi: 10.15439/2021f18.
- [3] Podbucki K., Possibilities and limitations of environment monitoring with usage of LiDAR scanner, *Przegląd Elektrotechniczny*, vol. 1, no. 1. Wydawnictwo SIGMA-NOT, sp. z.o.o., pp. 186–189, Jan. 04, 2022. doi: 10.15199/48.2022.01.40.
- [4] SAE International, Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles, SAE J3016 standard, 2021
- [5] Jencoe P., The AV from Algorithm to Acceleration, Under the Digital Hood: Adaptive Computing and AI for Autonomous Vehicles, *ElectronicDesign*, 2020, pp. 2-7
- [6] Suder, J.; Podbucki, K.; Marciniak, T.; Dąbrowski, A. Low Complexity Lane Detection Methods for Light Photometry System. *Electronics* 2021, 10, 1665. <https://doi.org/10.3390/electronics10141665>
- [7] Waveshare, JetRacer AI Kit, AI Racing Robot Powered by Jetson Nano, 17.03.2022, <https://www.waveshare.com/jetracer-ai-kit.htm>
- [8] NVIDIA Corporation, Jetson Nano Developer Kit, 17.03.2022, <https://developer.nvidia.com/embedded/jetson-nano-developer-kit>
- [9] Madrin F.P.; Rosenberger M.; Nestler R.; Dittich P.-G.; Notni G., The evaluation of CUDA performance on the Jetson Nano board for an image binarization task, *Proc. SPIE 11736, Real-Time Image Processing and Deep Learning 2021*, 117360G, 12 April 2021, <https://doi.org/10.1117/12.2586650>
- [10] Waveshare, IMX219 Camera Module, 160 degree FoV, 17.03.2022, <https://www.waveshare.com/imx219-d160.htm>
- [11] NVIDIA AI IOT, JetCard description, 22.03.2022, <https://github.com/NVIDIA-AI-IOT/jetcard>
- [12] Świdorski A., Wałęsa S., Monitoring of pedestrian crossings using an embedded system, Bachelor's thesis, Supervisor: Tomasz Marciniak, Auxiliary supervisor: Kacper Podbucki, Poznan University of Technology, 2022
- [13] S. Bianco, R. Cadene, L. Celona and P. Napoletano, "Benchmark Analysis of Representative Deep Neural Network Architectures," in *IEEE Access*, vol. 6, pp. 64270-64277, 2018, doi: 10.1109/ACCESS.2018.2877890.
- [14] Mathworks, Documentation of resnet18, 21.03.2022, <https://www.mathworks.com/help/deeplearning/ref/resnet18.html>
- [15] NVIDIA AI IOT, Benchmark of torch2trt function, 21.03.2022, <https://github.com/NVIDIA-AI-IOT/torch2trt>
- [16] Suder, J., Możliwości przetwarzania sekwencji wizyjnych w systemach wbudowanych, *Przegląd Elektrotechniczny*, No 01/2022, pp. 188-191, <https://doi.org/10.15199/48.2022.01.41>
- [17] Barreto-Cubero, A.J.; Gómez-Espinosa, A.; Escobedo Cabello, J.A.; Cuan-Urquiza, E.; Cruz-Ramírez, S.R. Sensor Data Fusion for a Mobile Robot Using Neural Networks. *Sensors* 2022, 22, 305. <https://doi.org/10.3390/s22010305>

Temporal Language Modeling for Short Text Document Classification with Transformers

Jakub Pokrywka, Filip Galiński

Adam Mickiewicz University,
 Faculty of Mathematics and Computer Science,
 Uniwersytetu Poznańskiego 4
 61-614 Poznań, Poland
 Email: {firstname.lastname}@amu.edu.pl

Abstract—Language models are typically trained on solely text data, not utilizing document timestamps, which are available in most internet corpora. In this paper, we examine the impact of incorporating timestamp into transformer language model in terms of downstream classification task and masked language modeling on 2 short texts corpora. We examine different timestamp components: day of the month, month, year, weekday. We test different methods of incorporating date into the model: prefixing date components into text input and adding trained date embeddings. Our study shows, that such a temporal language model performs better than a regular language model for both documents from training data time span and unseen time span. That holds true for classification and language modeling. Prefixing date components into text performs no worse than training special date components embeddings.

I. INTRODUCTION

MOST language models are trained solely on text data. Leveraging text domain, such as language [12] or style [10] into a language model may have a positive effect on it. Time of text authorship may be also considered as an input feature, but this poses specific challenges (and opportunities) as:

- time is continuous, whereas language is discrete, at any time moment, an event might change a language irreversibly and not trivial to combine time and language units both from the mathematical and practical standpoint;
- texts might reflect natural and social cycles (days, weeks, years, cyclical sport and political events);
- text content might be correlated with extralinguistic features, themselves correlating with time (e.g. air temperature).

Recently, the NLP community has started to use time as a feature in training and/or fine-tuning large neural models ([1], [16], [19]). Here, we analyze temporal language modeling in the context of two classification tasks in different timescales: Ireland News Headlines and Twitter Sentiment Analysis. We also incorporate date components other than year. We focus on examining different approaches to date incorporation (learnable embeddings, prefixing text) using periodic and non-periodic time features under a downstream classification task.

The contributions of this paper are as follows:

- two classification datasets were redefined in a common setup in which three time-related tasks are introduced: classification (possibly) using temporal metadata, predicting temporal metadata (as a regression task) and temporal language-modeling task (as a cloze task).
- we compared three methods for introducing temporal information into neural language models;
- we considered not only linear time, but also cycles such as years, weeks, and months;
- we measured the performance of RoBERTa [14] models in several setups on the two datasets (using different parts of the temporal information, and both fine-tuning and training from scratch);
- the relations between the temporal metadata, the texts and the results obtained were analyzed.

The datasets and source of our code are publicly available.

Generally, utilizing a date does not cost much effort, because many internet documents are available with a timestamp and it is possible to adapt existing models to new domain. Such temporal language models may contribute to:

- e-commerce search engines, e.g. users intention with short phrase "umbrella" may refer to umbrella protecting from a rain in the autumn or sun umbrella in the summer;
- other types of search engines, e.g. historical newspapers;
- OCR for historical documents.

II. DATASETS

Usually, text classification tasks do not incorporate time and other metadata. We suppose its impact is stronger for short texts due to shorter texts carrying less information. The time impact may be stronger for text, which may depend on people's mood or different interests. We carried out experiments with two large short-text classification datasets, where every sample is assigned a time stamp. One is spread over more than 20 years, the other ones — only 80 days. Both datasets are in English.

A. Ireland News

The dataset is available at Kaggle¹, its creator is Rohit Kulkarni. It consists of article headlines posted by the Irish

¹<https://www.kaggle.com/therohk/ireland-historical-news>

TABLE I: Categories count in datasets.

category	item			
	train	dev	test	test 20/21
news	603996	75963	75783	30278
business	162550	20330	20034	14477
sport	195384	24543	24346	13447
opinion	91697	11572	11528	8086
culture	67260	8525	8424	5643
life&style	65120	8093	8084	7188

Times newspaper. Each headline is accompanied by a timestamp and article category (text of an article is not included). There are six main categories: news, sport, opinion, business, culture, life&style. The datasets statistics are described in Table I. There are more fine-grained subcategories provided in the original dataset, but they vary over time, so we didn't make use of them in our experiments.

Timestamps range from 1996-01-01 to 2021-06-30. There are 1,611,495 such headlines in total.

We employed the date range from 1996-01-01 to 2019-12-31 for most of our experiments and created an additional test set, which consists of 2020-2021 years, which dates are non-overlapping with the rest of the dataset. We refer to this test set as **Ireland News 2020-2021**. The test set **Ireland News**, without year annotation, refers to time span from training data (1996-2019). Since train/dev/test split is not determined at the original dataset site, we assign each sample randomly to train/dev/test using the 80%/10%/10% split. This resulted in the 1,186,898 / 149,134 / 148,308 train/dev/test split. The average number of words in the dataset is 7.1 per headline.

B. Sentiment140

This sentiment analysis dataset is obtained and described in [2]. Since in the original dataset the train set contains 1,600,000 items (positive and negative tweets) and test set only 498 (positive, negative, and neutral tweets), we made significant modifications: neutral tweets were deleted from the test set, 100,000 random items were added to the test set, also a dev set was created by randomly selecting 100,000 samples from the train set. This resulted in the 1,400,000 / 100,000 / 100,359 train/dev/test split. Timestamps range from 2009-04-06 to 2009-06-25. The datasets set are balanced (~50% positive and ~50% negative tweets). The average number of words is 13.8 per item. Tweets are from users in different time zones. We take time local to the author of a tweet.

III. DATASETS ANALYSIS

The number of items per category differs in time. The distribution over days of month, months, years, weekdays in train datasets are presented in Figures 1 and 2 for, respectively, Sentiment140 and Ireland News. For the Sentiment140 dataset distribution over a year is not presented, since all items are from 2009. Mutual Information between presented factors and the class is given in Table V. In Ireland News, mutual information related to days of month and months is much lower than those of years and weekdays. In Sentiment140

mutual information is similar for days of month, months, and weekdays.

In both datasets, there are dependencies, which may be helpful for model performance. E.g. in Ireland News there are more sports texts on Friday and in Sentiment140 there are more negative texts on Wednesdays and Thursdays.

IV. TASKS

We created three tasks for each dataset: classification, 'fractional' year prediction, word gap prediction. Our main objective was to examine the impact of incorporating timestamps on text classification tasks. Fractional year prediction and word gap prediction tasks are mainly for analysis of the results in classification tasks.

We added timestamps in fractional-year form, which can be described by the following code:

```
days_in_year =
366 if year_is_leap_year else 365

fractional_year =
(year + (day_in_year-1+day) /
days_in_year )
```

Each item in our tasks is associated with a text, timestamp (day precision), fractional year, and category. Sample data is described in Table IV.

Each challenge for a given dataset uses the same train/dev/test split. The challenges are publicly available, courtesy of the site's owners, via the Gonito evaluation platform [3]. Source code of the challenge is available via the platform as well.

A. Classification

The task objective is to predict the headline category given text, date, and fractional year. The evaluation metric is simple accuracy. The challenges are available at: <https://gonito.net/challenge/ireland-news-headlines> (Ireland News) and <https://gonito.net/challenge/sentiment140> (Sentiment140). Dataset download and submission instructions are under the "How To" tab, source code is under the "All Entries" → catalog icon in each submission row.

B. Year prediction

The objective is to predict the year given the text. The metric is Root Mean Square Error (RMSE). The challenges are available at: <https://gonito.net/challenge/ireland-news-headlines-year-prediction> (Ireland News) and <https://gonito.net/challenge/sentiment140-year-prediction> (Sentiment140).

C. Word gap filling

The task objective is to predict a masked word, like in Masked Language Modeling, given text, date, fractional year. Word is defined by characters split by spaces. There is always exactly one masked word in each sample to

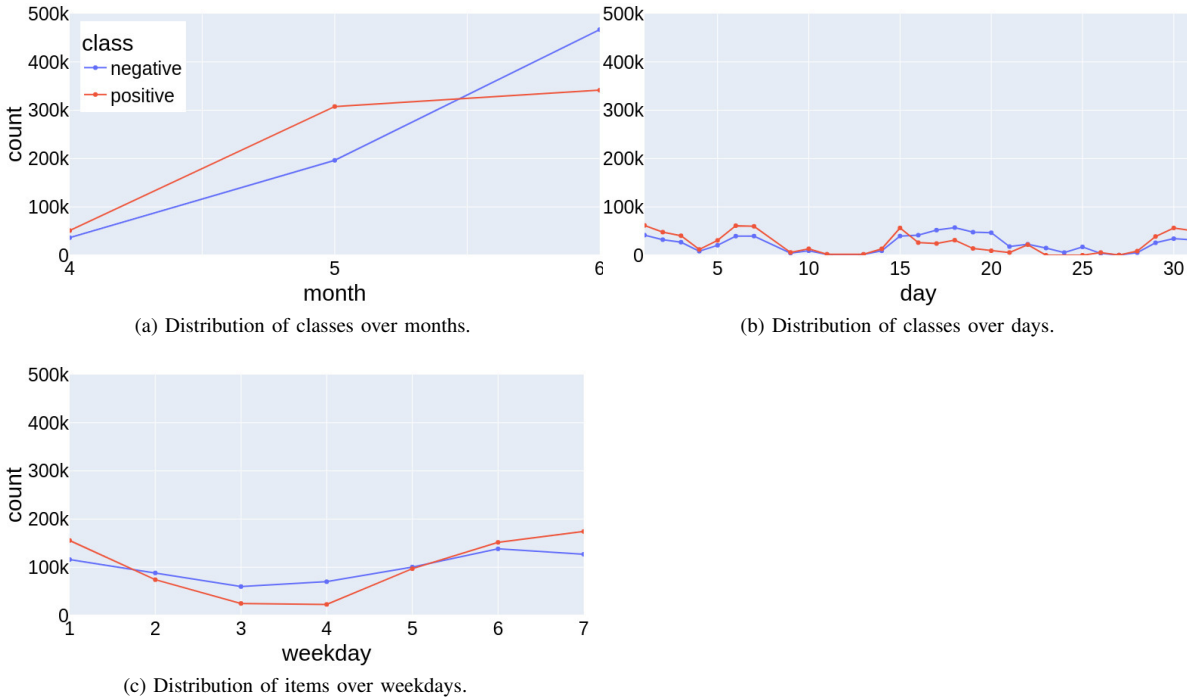


Fig. 1: Distribution of classes over date factors in Sentiment140 dataset. Distribution over year is not presented, since all items come from one year.

TABLE II: Samples from the Ireland News dataset. To check article-id visit www.irishtimes.com/article-id The article ID is not provided in the challenge.

fractional year	timestamp	text	category	article ID
2004.5082	20040705	Sudan claims it is disarming militias	news	1.1147721
2008.4426	20080611	Bluffer's guide to Euro 2008	sport	1.1218069
2017.1068	20170209	Gannon offers homes in Longview near Swords	life&style	1.2966726

predict. The metric is PerplexityHashed implemented in the GEval evaluation tool [4], which is a modified version of LogLossHashed as described by [5]. This metric ensures fair assessment disregarding model vocabulary. The challenges are available at: <https://gonito.net/challenge/ireland-news-headlines-word-gap> (Ireland News) and <https://gonito.net/challenge/sentiment140-word-gap> (Sentiment140).

V. METHODS

We used the RoBERTa model in the base version [14]. All models are described in this section. All code is publicly available via git commit hashes given in result tables.²

A. Regular Transformer as a baseline

The baseline is a regular RoBERTa with no temporal information. We refer to this method as noDate in result tables.

²Reference codes to repositories stored at Gonito.net [3] are given in curly brackets. Such a repository may be also accessed by going to <http://gonito.net/q> and entering the code there.

B. Temporal Transformer

We selected the following temporal information: year, month, day of the month (day), weekday. All of them are incorporated in our temporal models. We experimented with 3 ways of including temporal information into RoBERTa models. The first two involve slight RoBERTa model architecture changes and training new embeddings during RoBERTa training. The third one is only input data modification. They are described below.

1) *Date as embeddings added to every input token:* Temporal embeddings are added to every input token as: $embedding = token_emb + pos_emb + year_emb + month_emb + monthday_emb + weekday_emb$ for each $token_pos$. We refer to this method as addedEmbDate in result tables.

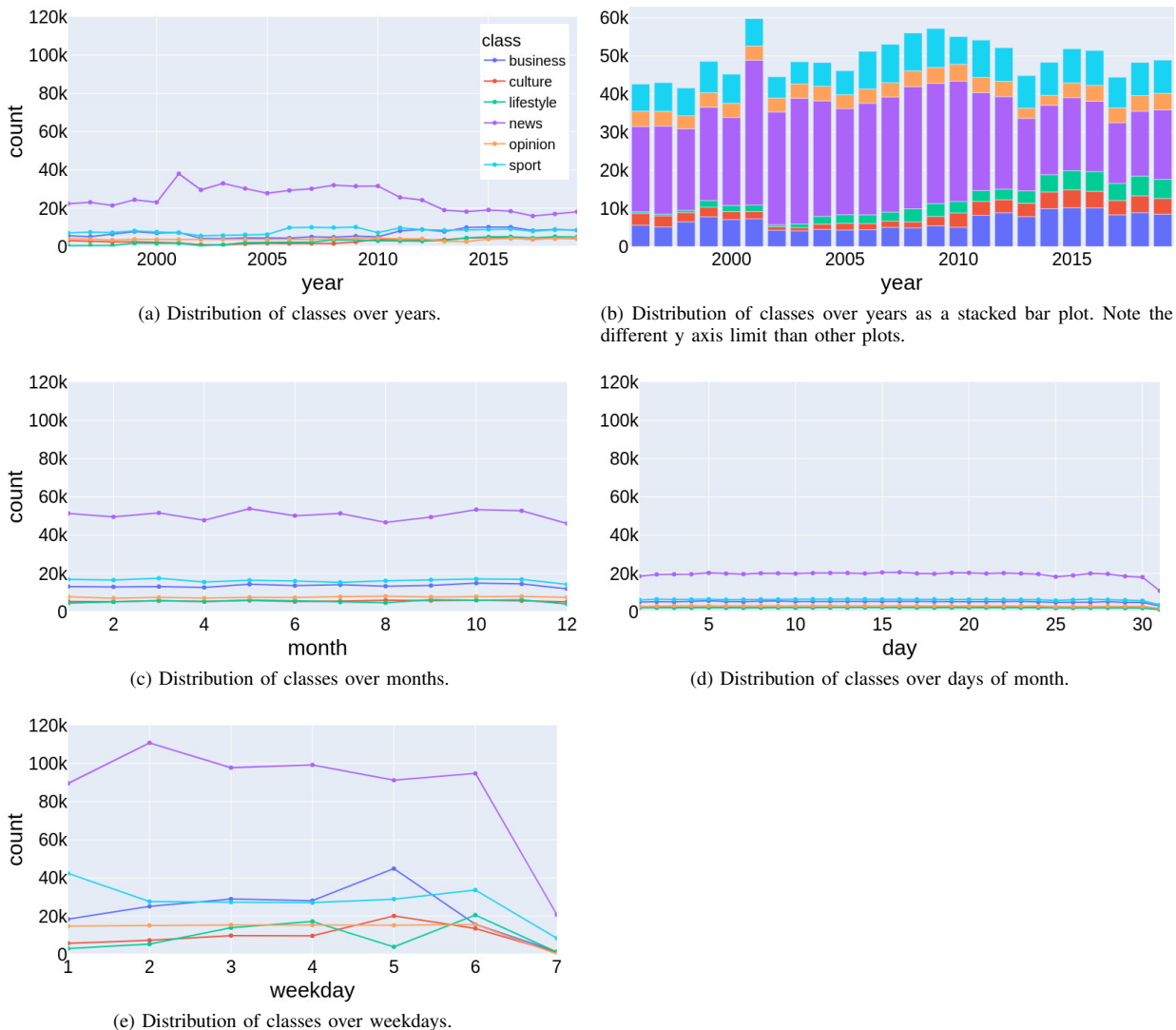


Fig. 2: Distribution of items over date factors in Ireland News dataset.

2) *Date as stacked embeddings*: Temporal embeddings are stacked at the beginning of the input sequence, as:

$$emb = \begin{cases} year_emb & \text{if } token_pos = 1 \\ month_emb & \text{if } token_pos = 2 \\ month_emb & \text{if } token_pos = 3 \\ weekday_emb & \text{if } token_pos = 4 \\ token_emb+ & \\ pos_emb & \text{otherwise} \end{cases}$$

Where all tokens are shifted 4 positions to the right, so first text token is on $token_pos = 5$. We refer to this method as `stackedEmbDate` in result tables.

3) *Date as regular text*: We only modify text input of model by adding temporal information with prefixes, so item with date `20040705` and text `Sudan claims it is disarming militias` is combined to text `year: 2004`

`month: 7 day: 5 weekday: 1 Sudan claims it is disarming militias.`

VI. EXPERIMENTS

A. Classification

We carried out experiments with text classification using all presented models. RoBERTa was finetuned and trained from pretrained checkpoints (which we refer to as `pretrained`) and with randomly initialized weights (which we refer to as `from scratch`). The only training objective is the classification task. We report the results in Table IV.

We examined the impact on classification by each date factor. Since all temporal data incorporation methods yield similar results, we chose the regular text date incorporation method due to ease of its use (only text modification with no architecture changes). The results are presented in Table V. To examine this model conditioned by different prefixes we

TABLE III: Model roberta-pretrained-textDate predictions depending on a given date in a development dataset. If a date is represented by a dash, it is not prefixed to the model, bolded dates are as they occur actually in the dataset, not bolded are random. The examples are cherry-picked. To check article-id visit www.irishtimes.com/article-id The article ID is not provided in the challenge.

text	article ID	timestamp	actual	prediction
New bridge for Calzaghe to cross	1.914946	20080419 Sat.	sport	sport
New bridge for Calzaghe to cross	1.914946	20130307 Thu.	-	life&style
New bridge for Calzaghe to cross	1.914946	-	-	news
Sydney stereotypes	1.1102371	20000913 Wed.	sport	sport
Sydney stereotypes	1.1102371	20110422 Fri.	-	opinion
Sydney stereotypes	1.1102371	-	-	sport
Róisín Meets... comedian Mario Rosenstock	1.2463531	20151212 Sat.	life&style	life&style
Róisín Meets... comedian Mario Rosenstock	1.2463531	20040725 Sun.	-	news
Róisín Meets... comedian Mario Rosenstock	1.2463531	-	-	news

TABLE IV: Classification results. Different date incorporation into model. Acc stands for accuracy. The bold results are best in its category (without and with external data).

method	Ireland News		Sentiment140	
	acc	gonito	acc	gonito
most frequent from train	51.10	{161712}	49.88	{b4b180}
roberta-pretrained-noDate	82.35	{daaaf9}	89.27	{a8d1b7}
roberta-pretrained-stackedEmbDate	87.65	{9e041f}	91.16	{252c0c}
roberta-pretrained-addedEmbdate	86.82	{cede76}	91.04	{aa28dc}
roberta-pretrained-textDate	87.84	{7c52ed}	91.13	{688320}
roberta-scratch-noDate	77.88	{0798d5}	83.38	{e984db}
roberta-scratch-stackedEmbDate	83.24	{74efba}	86.18	{e3ff3e}
roberta-scratch-addedEmbdate	81.96	{587033}	85.47	{1c122b}
roberta-scratch-textDate	83.16	{413f72}	86.02	{d969ca}

TABLE V: Classification accuracy results. Different date elements included. Acc stands for accuracy. MI stands for Mutual Information between a class and a date factor. MI for Sentiment140 between year and class equals 0, because there is only 2009 year in the dataset.

method	Ireland News			Sentiment140		
	Acc	Gonito	MI(1e-5)	Acc	Gonito	MI(1e-3)
roberta-pretrained-noDate	82.35	{daaaf9}	-	89.27	{a8d1b7}	-
roberta-pretrained-textDate	87.84	{7c52ed}	-	91.13	{688320}	-
roberta-pretrained-textDay	82.66	{ca5340}	9	90.16	{2c2d07}	58
roberta-pretrained-textMonth	82.72	{3d5bb6}	61	89.59	{64cc1b}	16
roberta-pretrained-textYear	85.90	{893bbe}	3354	89.32	{be6d55}	0
roberta-pretrained-textWeekday	84.46	{daf69a}	3127	89.60	{8abd71}	19

TABLE VI: Roberta-pretrained-textDate classification on development set result. All results comes from the same model, the only difference is the prefix construction. Prefix is a standard model mode, no-prefix is a mode where no date is prefixed, and random-prefixed stands for a mode, where the date prefix comes from random date 1996-01-01 to 2021-06-30.

model	dev acc
prefix	87.97
no-prefix	78.38
random-prefix	73.97

checked its performance with no prefix and random prefix settings. Results are in Table VI and Table VII. The samples

from different prefix settings are provided in Table IV.

To check model degradation, we made an inference on Ireland News test set from years 2020-2021. This is a time span later than training data, which comes from 1996-2019. The results are in Table VIII.

The impact of train dataset size is presented in Figure 3.

B. Year prediction

We choose two methods for year prediction. The first is a baseline using term frequency-inverse document frequency (TF-IDF) with logistic regression. The second is averaging all output embeddings of RoBERTa and feeding to linear regression (LR) layer. Both RoBERTa and linear regression weights are tuned during training. In both methods, the minimum (maximum) output is limited to the minimum (maximum)

TABLE VII: Classification improvement due to prefixing on roberta-pretrained-textDate model. All results comes from the same model, naming convention comes from Table VI.

dev set percentage	
accurate on both prefix and no-prefix	75.14
accurate on prefix, but not on no-prefix	12.83
accurate on no-prefix, but not on prefix	3.19
not accurate on prefix, nor on no-prefix	9.84

TABLE VIII: Classification accuracy results. Test set (years 2020-2021) comes from other time span than training set (years 1996-2019).

method	Ireland News (2020/21)	
	acc	gonito
most frequent	38.27	{953311}
roberta-pretr.-noDate	85.99	{e684b3}
roberta-pretr.-textDate	87.79	{5fba22}
roberta-pretr.-textYear	87.49	{8d5ad4}

fractional year found in the datasets. The results are presented in Table IX, along with a null-model baseline using the mean fractional year from the training set as the prediction for each data point.

C. Word gap filling

RoBERTa was finetuned and trained from a pretrained checkpoint and with randomly initialized weights. The training objective is Masked Language Modeling. Only prepending data to the input was considered as a method for introducing the data. See Table X.

VII. DISCUSSION

For both datasets including dates into RoBERTa models raises the accuracy score. This stands true for pretrained and randomly initialized models. Stacked embedding and date incorporation as a text give a similar result and both are slightly better than the method of adding embeddings to every input token. It's easier to modify input text than modify model architecture, hence we recommend embedding date by prefixing input texts. The greater mutual information is between each factor and class factor, the more the model gains in accuracy score. The model trained with a date prefix performs well, only when the prefix is provided. There is no gain from date prefixing for a 1k documents train dataset and the gain is constant over 100k documents train dataset. Predicting fractional year is difficult in both datasets because all models perform not much better than baseline. We hypothesize this is a reason why classification benefits from date metadata, since adding strongly correlated factors (like a date to text in this case) would not bring information gain.

The temporal models perform better also for test sets from unseen years. To our surprise, day of the month, month, weekday, year incorporation into model performs only marginally better than incorporation only year for Ireland News 2020-2021 dataset.

In pretrained models, date incorporation slightly lowers perplexity. Models with randomly initialized weights benefit hugely from date incorporation.

VIII. RELATED WORK

There are several studies concerning language model degradation over time and adaptation to newer data [13], [17], [6]. [7] focused especially on text classification. They considered years as well as cyclical intervals (e.g., January-March). Their method was to train separate models for different time spans. [8] proposed method based on using discrete multiple temporal word embeddings based on time domains for document classification using recurrent neural networks. [9] developed model-agnostic timed dependent embedding representation for time and evaluated on recurrent neural networks across various tasks. [1] introduced temporal T5 language model, where a year was prefixed into text input and finetuned on temporal data. The experiments focused on knowledge extraction from language models and showed their method performs better in terms of language modeling and question answering than T5 language model with no prefixed year. [19] incorporated both geographical and time data into a transformer model for a QA task employing year as well as month and day. [16] prefixed year for semantic change detection. Additionally, the authors proposed the training objective of masking year information during model training. However, both [1], [16] use only year metadata, in contrast to our study, where we also days of month, months, weekdays are taken into consideration. [18] trained an SVM model to predict the date of text as a classification problem and [11] use approach of neologism based approach. Very recently [15] released temporal NLP challenges based on a large corpus of historic texts but didn't include downstream tasks, such as classification. The corpus consists of texts covering over 100 years. They trained from scratch and fine-tuned temporal RoBERTa models based on day of month, months, weekdays, and year as a prefixed text. They proved that temporal language models perform better than standard language models.

IX. CONCLUSION

Transformer models benefit from temporal information data in classification tasks for short texts. We have proved that it's not only true for a year, but also other date factors, such as weekday, day of the month, and month. The greater the mutual information between a factor and a class, the greater the benefit. The result is important, because day of the month, month, weekday factors don't outdate after model training

TABLE IX: Fractional year prediction results, RMSE is for root-mean-square error, MAE – mean absolute error, LR – linear regression.

method	Ireland News			Sentiment140		
	RMSE	MAE	Gonito	RMSE	MAE	gonito
mean from train	6.76426	5.80722	{0b0e9c}	0.04674	0.03396	{4856c5}
TF-IDF + LR	5.32491	4.27185	{2226fb}	0.04917	0.03635	{579c8f}
RoBERTa + LR head	4.53676	3.38758	{632b5d}	0.04469	0.03289	{349e5b}
RoBERTa from scratch + LR head	4.51179	3.35951	{be0106}	0.04526	0.03222	{b672ee}

TABLE X: Word gap prediction results. Ppl hashed stands for perplexity hashed.

method	Ireland News		Sentiment140	
	ppl hashed	gonito	ppl hashed	gonito
equal probability	1024.0	{6bd5a8}	1024.0	{3de230}
RoBERTa from scratch	90.8	{9ac479}	51.0	{f0f343}
RoBERTa from scratch with time	46.0	{dc75a7}	46.1	{ddf16f}
RoBERTa no fine-tuning	51.0	{f0f343}	66.2	{e625c6}
RoBERTa fine-tuned	23.3	{42793a}	34.6	{a365da}
RoBERTa fine-tuned with time	21.6	{cfaf6c}	33.6	{37bd6e}

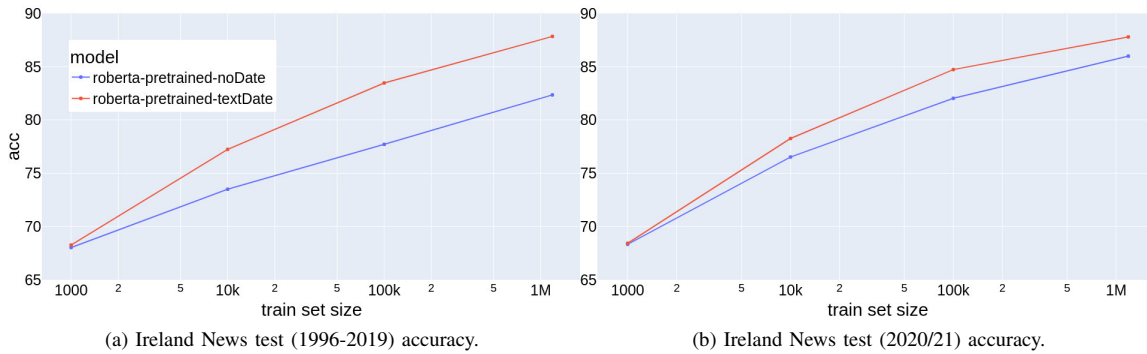


Fig. 3: Test set accuracy varying on train dataset size for model with and without date incorporation.

due to its cyclical nature, differently to year, which is linear. The best and simplest method for temporal data incorporation seems to be input text modification.

REFERENCES

[1] B. Dhingra, J. R. Cole, J. M. Eisenschlos, D. Gillick, J. Eisenstein, and W. W. Cohen. Time-aware language models as temporal knowledge bases, 2021.

[2] A. Go, R. Bhayani, and L. Huang. Twitter sentiment classification using distant supervision. *Processing*, 150, 2009.

[3] F. Graliński, R. Jaworski, Ł. Borchmann, and P. Wierchoń. Gonito.net – open platform for research competition, cooperation and reproducibility. In A. Branco, N. Calzolari, and K. Choukri, editors, *Proceedings of the 4REAL Workshop: Workshop on Research Results Reproducibility and Resources Citation in Science and Technology of Language*, pages 13–20. 2016.

[4] F. Graliński, A. Wróblewska, T. Stanisławek, K. Grabowski, and T. Górecki. GEval: Tool for debugging NLP datasets and models. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 254–262, Florence, Italy, 2019. Association for Computational Linguistics.

[5] F. Graliński. (Temporal) language models as a competitive challenge. In Z. Vetulani and P. Paroubek, editors, *Proceedings of the 8th Language & Technology Conference*, pages 141–146. Fundacja Uniwersytetu im. Adama Mickiewicza w Poznaniu, 2017.

[6] S. A. Hombaiyah, T. Chen, M. Zhang, M. Bendersky, and M. Najork. Dynamic language models for continuously evolving content. *ArXiv preprint, abs/2106.06297*, 2021.

[7] X. Huang and M. J. Paul. Examining temporality in document classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 694–699, Melbourne, Australia, 2018. Association for Computational Linguistics.

[8] X. Huang and M. J. Paul. Neural temporality adaptation for document classification: Diachronic word embeddings and domain adaptation models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4113–4123, Florence, Italy, 2019. Association for Computational Linguistics.

[9] S. M. Kazemi, R. Goel, S. Eghbali, J. Ramanan, J. Sahota, S. Thakur, S. Wu, C. Smyth, P. Poupart, and M. A. Brubaker. Time2vec: Learning a vector representation of time. *ArXiv, abs/1907.05321*, 2019.

[10] N. S. Keskar, B. McCann, L. R. Varshney, C. Xiong, and R. Socher. Ctrl: A conditional transformer language model for controllable generation. *ArXiv, abs/1909.05858*, 2019.

[11] V. Kulkarni, Y. Tian, P. Dandiwalwa, and S. Skiena. Simple neologism based domain independent models to predict year of authorship. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 202–212, Santa Fe, New Mexico, USA, 2018. Association for Computational Linguistics.

[12] G. Lample and A. Conneau. Cross-lingual language model pretraining. In *NeurIPS*, 2019.

[13] A. Lazaridou, A. Kuncoro, E. Gribovskaya, D. Agrawal, A. Liska, T. Terzi, M. Gimenez, C. de Masson d’Autume, S. Ruder, D. Yogatama,

- K. Cao, T. Kociský, S. Young, and P. Blunsom. Pitfalls of static language modelling. *ArXiv*, abs/2102.01951, 2021.
- [14] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. RoBERTa: A robustly optimized BERT pretraining approach. *ArXiv preprint*, abs/1907.11692, 2019.
- [15] J. Pokrywka, F. Graliński, K. Jassem, K. Kaczmarek, K. Jurkiewicz, and P. Wierzchoń. Challenging America: Modeling language in longer time scales. *Findings of North American Chapter of the Association for Computational Linguistics*, 2022. forthcoming.
- [16] G. D. Rosin, I. Guy, and K. Radinsky. Time masking for temporal language models, 2021.
- [17] P. Röttger and J. Pierrehumbert. Temporal adaptation of BERT and performance on downstream document classification: Insights from social media. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2400–2412, Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics.
- [18] T. Szymanski and G. Lynch. UCD : Diachronic text classification with character, word, and syntactic n-grams. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 879–883, Denver, Colorado, 2015. Association for Computational Linguistics.
- [19] M. Zhang and E. Choi. SituatedQA: Incorporating extra-linguistic contexts into QA. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7371–7387, Online and Punta Cana, Dominican Republic, 2021. Association for Computational Linguistics.

An End-to-end Machine Learning System for Mitigating Checkout Abandonment in E-Commerce

Md Rifatul Islam Rifat¹, Md Nur Amin², Mahmud Hasan Munna³, and Abdullah Al Imran⁴

^{1,3}Rajshahi University of Engineering & Technology, Rajshahi, Bangladesh
Email: irifat.ruet@gmail.com, munna.ete15@gmail.com

²University Jean Monnet, Saint Etienne, France
Email: nuramin.aiub@gmail.com

⁴American International University-Bangladesh, Dhaka, Bangladesh
Email: abdalimran@gmail.com

Abstract—Electronic Commerce (E-Commerce) has become one of the most significant consumer-facing tech industries in recent years. This industry has considerably enhanced people’s lives by allowing them to shop online from the comfort of their own homes. Despite the fact that many people are accustomed to online shopping, e-commerce merchants are facing a significant problem, a high percentage of checkout abandonment. In this study, we have proposed an end-to-end Machine Learning (ML) system that will assist the merchant to minimize the rate of checkout abandonment with proper decision making and strategy. As a part of the system, we developed a robust ML model that predicts if someone will checkout the products added to the cart based on the customer’s activity. Our system also provides the merchants with the opportunity to explore the underlying reasons for each single prediction output. This will indisputably help the online merchants in business growth and effective stock management.

Index Terms—E-Commerce, Checkout Prediction, Checkout Abandonment, Decision Support, Explainable AI (XAI), LIME

I. INTRODUCTION

AS we are living in an era where digitalization and technology are evolving day by day, our dependency on the internet has noticeably increased. E-commerce has made shopping easy and safe for all internet users all over the world. People nowadays prefer exploring websites to find their daily needs rather than walking around shopping malls, supermarkets, and shops. They do not have to take the hassle of finding a product and waiting for a long billing queue, which makes purchasing simple and quick. On the contrary, e-commerce also makes it easier for companies to reach out to new customers all over the world. A report from Statista [1] shows that global sales have jumped from 1,336 billion to 5,542 billion USD in the last 6 years in the e-commerce industry. In the near future, undoubtedly the dependency on online shopping will increase significantly.

Recently, the online retailers are encountering numerous business challenges such as the lack of trust, customer churn, product return and so on. With the rapid technological advancement in the data science domain, researchers have already started to solve these type of problems by utilizing data science approaches. Some of the existing research works

are related to the product review classification such as the authors in [2] proposed DNN networks to train a classifier for identifying the product quality from product reviews. One of the major problems in e-commerce industry is the return of the product. In [3] and [4], the researchers tried to address this issue by using different predictive modeling techniques.

Apart from these, one of the most common business challenges in the e-commerce industry is high checkout abandonment rate. According to the study of Baymard Institute, a research institute in Denmark, the average checkout abandonment rate is 69.82% [5]. Also during the COVID-19 pandemic, online shopping behaviors have been significantly changed. When individuals browse an online store for a particular product, as a natural consequence, they often add many additional items to their cart. Among the added items in the cart, some are in need and the others may be their favorite but are not in great demand, and most of the time the majority of them are never checked out which results in high checkout abandonment rate. However, very few researchers have contributed to solve the issues related to online shopping carts. Jian et al [6] proposed a framework to predict buyers’ repurchases intention from the cart information. In another study [7], the author built a recommendation system using the shopping cart information.

In this research, we have tried to address the aforementioned business problem of the e-commerce industry and proposed an end-to-end ML system that will automatically perform all the steps such as data collection, transformation, preprocessing, statistical analysis, and predictive analytics. In the case of predictive analytics, we have conducted an extensive experiment and found CatBoost as the outperforming approach that predicts the checkout possibility of users with the highest accuracy ($=0.76$) and precision ($=0.694$). Moreover, we have applied a model agnostic local explanation approach for the explainability that will help the e-commerce merchants to analyze how every single customer gets influenced by different factors.

Our contribution to this study can be considered from two perspectives: one from a research standpoint, and the other from a commercial standpoint. The research perspective is that

no previous study attempted to solve this particular problem and proposed any end-to-end ML based solution. On the other hand, if business people conduct targeted marketing or apply other business strategies to consumers who are most likely to purchase the products in the cart, then the sales will be increased. Apart from that, the prediction will assist the merchant in maintaining effective stock management as well. Indisputably, the combination of statistical insights, predictive output, and local explanation aid the seller in developing proper strategies for business growth.

II. PROPOSED SOLUTION

As a solution to the checkout abandonment issue in e-commerce business, in this phase, we have proposed an end-to-end system shown in Fig. 1.

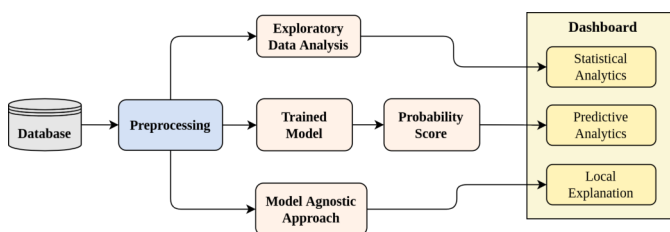


Fig. 1. System diagram

From Fig. 1 it can be observed that our proposed system takes raw data from the database and outputs analytical and predictive insights to a dashboard. Development of this system is composed of several steps: data understanding, preprocessing, exploratory data analysis, data modeling, and a model-agnostic approach.

To conduct the experiment, a large real dataset has been collected from a prominent SaaS platform that integrates with online stores to track behavior in real-time. Our dataset contains 27 features in total, where 21 features are numerical and the other 6 features are categorical. Table I and Table II shows the description of all the numeric and categorical features respectively. Among all the instances(=28410), 55% instances belong to class 0 ('not checked out') and the other 45% belongs to class 1 ('checked out') which indicates that the dataset is almost balanced.

After collecting the data, in the preprocessing phase, we have applied the James-Stein encoder to convert the categorical features into informative numerical representations. The mathematical expression of the James-Stein encoder is as follows:

$$JS_i = (1 - B) \times mean(y_i) + B \times mean(y) \quad (1)$$

Where

$$B = \frac{var(y_i)}{var(y_i) + var(y)} \quad (2)$$

The idea of the James-Stein encoder is to shrink the category's mean target towards a more median average.

As most of the features do not fall into a Gaussian distribution, in this experiment, we have applied min-max scaling to scale the features from 0 to 1.

TABLE I
DATA DESCRIPTION OF NUMERICAL FEATURES

Feature Name	Description
visited_cart	How many times did the user visit the "Cart" page so far in the current visit.
total_add_cart	How many products did the user add to the shopping cart.
total_clicked_products	How many times did the user click on products or visit the "Product Page" of products.
session_length_steps	Length of the visit in steps/events.
session_length_sec	Length of the visit in seconds.
mean_viewed_price	Mean price of clicked products.
max_viewed_price	Max price of clicked products.
min_viewed_price	Min price of clicked products.
sum_viewed_price	Sum price of clicked products.
total_orders	Total orders the user checked out so far.
total_purchased_sum	Total sum of orders the user checked out so far.
total_visits	Total previous visits of the users.
returning_visitor	Is it a new or returning user.
customer_since_days	Total minutes since user's first visit.
cart_sum	The sum of the shopping cart the user has/sees it right now.
cart_total_prd	Total products the user has in the shopping cart.
cart_total_sale_prd	Total discounted products the user has in the shopping cart.
cart_total_prd_in_sale	The ratio of discounted products vs non discounted products in the cart.
cart_total_saved	The amount of money the user is saving due to the presence of sale products in the cart.
cart_sum_without_discount	Total sum of the cart on original price of the products.
cart_total_saved_%	Total percentage of saved money in the cart.

TABLE II
DATA DESCRIPTION OF CATEGORICAL VARIABLES

Feature Name	Description
landing_page	The page where the user started the visit.
week_day	Day of the week
origin	From which origin did the user land on the store.
utm_source	From which source did the user land on the store.
utm_medium	From which medium did the user land on the store.
device	The user's device type.

Then, during the data modeling phase, we have followed a proper and systematic workflow that has been illustrated in Fig. 2. This workflow has been started just the completion of the data preprocessing steps. At first, we have segregated the entire processed dataset into training (=80%) and validation (=20%) data to eliminate biases during the model evaluation phase. Then, in the second stage, we have chosen the ML algorithms based on the objectives as well as the characteristics of the data. In this study, we have applied 5 SOTA algorithms such as XGBoost [9], LightGBM [10], CatBoost [11], mGBDTs [13], and TabNet [12] not only targeting the best prediction output but also with a special focus on interpreting the models. To create a baseline performance, we have also included a DNN model.

Then in the evaluation phase, we have applied 5 evaluation metrics namely Accuracy, Precision, Recall, $f_{0.5}$ -score, and ROC-AUC. The fourth and most important step is to tune the hyperparameters of the model very attentively to obtain the best prediction output without over-fitting or under-fitting. For

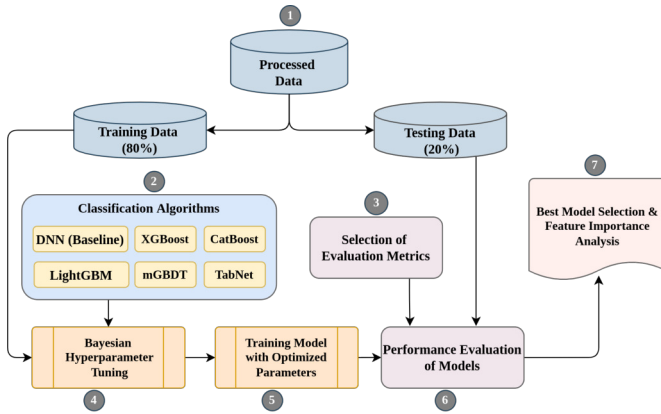


Fig. 2. Flow-diagram of Data Modeling

TABLE III
MODEL PERFORMANCE ON TRAINING AND TESTING DATA.

		ROC -AUC	Precision	Recall	$f_{0.5}$ Score	Accu- racy
DNN	Train	0.565	0.586	0.972	0.858	0.606
	Test	0.565	0.585	0.971	0.636	0.606
XGBoost	Train	0.798	0.742	0.778	0.771	0.802
	Test	0.753	0.689	0.729	0.720	0.758
CatBoost	Train	0.804	0.755	0.779	0.774	0.809
	Test	0.755	0.694	0.725	0.719	0.760
LightGBM	Train	0.782	0.724	0.759	0.752	0.786
	Test	0.746	0.680	0.723	0.714	0.751
TabNet	Train	0.757	0.688	0.746	0.733	0.739
	Test	0.730	0.655	0.722	0.702	0.723
mGBDT	Train	0.793	0.743	0.764	0.760	0.798
	Test	0.733	0.673	0.694	0.690	0.741

tuning the hyperparameters, the Bayesian approach has been chosen in this experiment since it takes less time compared to others to find the best set of parameters and improves generalization performance on the test data. After getting the optimized hyperparameters for all of the models, in the fifth stage, we have trained our models using the training dataset. With the completion of the training phase, we went on to the sixth stage, in which we have utilized the trained models to make predictions on the unseen validation dataset and track their performance against each evaluation metric. Then, in the following step, we have compared the performance among the models to figure out the best model for this particular business problem.

III. BEST MODEL AND FEATURES SELECTION

In this phase, we have divided our analysis into two parts. We have started the analysis by explaining the performance of each model and selecting the best model from their comparison. Finally, we have explored the top features of our best model to find insight that can help to make important business decisions.

Table III shows both of the training and testing results for each of the models of our experiment. From the Table III, it can be observed that our obtained test results are very close to the training results, which indicates that the model has learned

the underlying patterns well from the data without over-fitting. By considering the business problem we were trying to solve, we have mainly focused on Precision, $f_{0.5}$ -score, and Accuracy. Significantly, it has been appeared that all the models have been performed better than our baseline DNN model in terms of Precision, $f_{0.5}$ -score, and Accuracy. Although TabNet pretrained model performs well for tabular data, in our case it fails to outperform the tree-based models. Similarly, the mGBDT fails to outperform the other non-differentiable boosting algorithms. All of the tree-based boosting algorithms: XGBoost, Catboost, and LightGBM have yielded the results close to each other. Comparing the results of each models, it can be seen that CatBoost has consistently outperformed all other models with accuracy (=0.76) and precision (= 0.694). As the conclusion of all comparisons, we chose CatBoost as the best performing model for checkout prediction.

Furthermore, we have extracted the most effective 5 features from the CatBoost classifier. Table IV shows the best features in descending order based on the feature importance.

TABLE IV
FEATURE IMPORTANCE (TOP 5) OF CATBOOST CLASSIFIER

Rank	Features	Importance(%)
1	total visits	14.09
2	customer since days	13.56
3	total purchased sum	6.46
4	max viewed price	5.90
5	min viewed price	5.75

TABLE V
EXAMPLES FROM VALIDATION DATA FOR LOCAL EXPLANATIONS.

Features	Instance 1	Instance 2
visited_cart	1	1
total_add_cart	1	2
total_clicked_products	6	2
session_length_steps	38	5
session_length_sec	1564.05	111.54
mean_viewed_price	213.62	48.9
max_viewed_price	239	48.9
min_viewed_price	169.9	48.9
sum_viewed_price	1281.7	97.8
total_orders	0	1
total_purchased_sum	0	185
total_visits	10	1
returning_visitor	1	1
customer_since_days	0	1704.9
cart_sum	239	97.8
cart_total_prd	1	2
cart_total_sale_prd	0	2
cart_total_prd_in_sale	0	100
cart_total_saved	0	42
cart_sum_without_discount	239	139.8
cart_total_saved_%	0	30.04
landing_page	home page	home page
week_day	wednesday	thursday
origin	google	google
utm_source	unknown	google
utm_medium	unknown	cpc
device	mobile	desktop
checkout_status	not-checkout	checkout

IV. MODEL EXPLANATION AND DECISION SUPPORT

To make our model's decision more transparent, in this study, we have built a support system using the model agnostic local explanation technique, LIME [14]. This method will assist us in comprehending the factors that influence a complex black-box model around a single instance of interest.

In the following Table V, we have taken two instances from the unseen validation data for local explanations. Instance 1 has a checkout-abandonment status, and Instance 2 has a checkout status. The prediction and explanations for these examples can be found in Fig. 3 and 4.

Fig 3 illustrates the decision rules and feature significance based on which the CatBoost model made the decision for instance 1, which was actually abandoning the checkout after adding items to the cart. We can observe that the model predicts with a 95% probability that this person will abandon the checkout. Also, the aforementioned 3 most important features: "total visits", "customer since days", and "total purchased sum" significantly influenced the model to decide in favor of checkout abandonment.

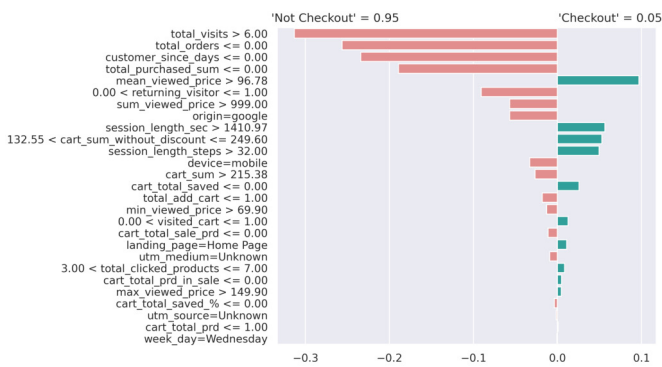


Fig. 3. Explanation of instance 1.

In Fig 4, we can observe that the model predicts with 99% probability that instance 2 will checkout the added item which is actually correct. From the value of "customer since days" feature, it is obvious that being the old customer gave the model confidence that instance 2 will checkout the products.

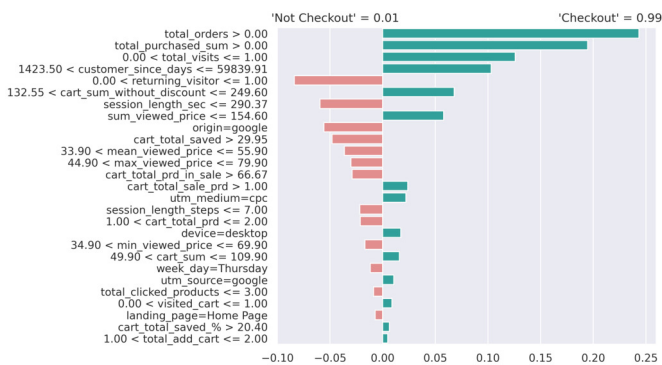


Fig. 4. Explanation of instance 2.

The aforementioned illustrations of local explanations of predictions for individual examples can immensely assist business analysts or decision-makers in making meaningful and unbiased decisions. Especially, this explanation technique can especially assist in making decisions in confusing situations where it is difficult to decide whether a person will checkout the items from the cart or not.

V. CONCLUSION

In this study, we have aimed to minimize the checkout abandonment rate, which is a key concern in modern e-commerce business, by proposing an end-to-end system. One of the most important components of the system is the ML model that predicts the probability of checkout abandonment for each of the customer. Also, it provides the explanation of the decision taken by the model for further business support. In case of predictive analytics, the CatBoost was found as the best performer with 0.694 (Precision), 0.719 ($f_{0.5}$ -score), and 0.76 (Accuracy). For reliable decision making with additional support, we have integrated the LIME, that interprets the model's output as well as extract the decision rules.

REFERENCES

- [1] Staista, <https://www.statista.com/statistics/379046/worldwide-retail-e-commerce-sales>. Last accessed 8 Apr 2022
- [2] M. H. Munna, M. R. I. Rifat and A. S. M. Badrudduza, "Sentiment Analysis and Product Review Classification in E-commerce Platform," *2020 23rd International Conference on Computer and Information Technology (ICCIT)*, 2020, pp. 1-6, DOI: 10.1109/ICCIT51783.2020.9392710.
- [3] Al Imran, A. and Amin, M.N., 2020. Predicting the return of orders in the e-commerce industry accompanying with model interpretation. *Procedia Computer Science*, 176, pp.1170-1179. DOI: 10.1016/j.procs.2020.09.113
- [4] Urbanke, P., Kranz, J. and Kolbe, L., 2015. Predicting product returns in e-commerce: the contribution of mahalanobis feature extraction.
- [5] Baymard Institute, <https://baymard.com/lists/cart-abandonment-rate>. Last accessed 8 Apr 2022
- [6] Mou, J., Cohen, J., Dou, Y. and Zhang, B., 2017. Predicting Buyers' repurchase Intentions in Cross-Border E-Commerce: A Valence Framework Perspective.
- [7] Budnikas, G., 2015. Computerised recommendations on e-transaction finalisation by means of machine learning. *Statistics in Transition. New Series*, 16(2), pp.309-322.
- [8] Cox, N.J., 2010. Speaking Stata: The limits of sample skewness and kurtosis. *The Stata Journal*, 10(3), pp.482-495. DOI: 10.1177/1536867X1001000311
- [9] Chen, T. and Guestrin, C., 2016, August. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785-794). DOI: 10.1145/2939672.2939785
- [10] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q. and Liu, T.Y., 2017. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30.
- [11] Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A.V. and Gulin, A., 2018. CatBoost: unbiased boosting with categorical features. *Advances in neural information processing systems*, 31.
- [12] Arik, S.Ö. and Pfister, T., 2021, May. Tabnet: Attentive interpretable tabular learning. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 35, No. 8, pp. 6679-6687).
- [13] Feng, Ji, Yang Yu, and Zhi-Hua Zhou. "Multi-layered gradient boosting decision trees." *Advances in neural information processing systems 31* (2018).
- [14] Ribeiro, M.T., Singh, S., Guestrin, C.: "Why should i trust you?" explaining the predictions of any classifier. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1135-1144 (2016) DOI: 10.1145/2939672.2939778

Reinforcement Learning for on-line Sequence Transformation

Grzegorz Rypeś, Łukasz Lepak, Paweł Wawrzyński
Warsaw University of Technology, Institute of Computer Science, Warsaw, Poland
{grzegorz.rypec.stud, lukasz.lepak.dokt, pawel.wawrzynski}@pw.edu.pl

Abstract—In simultaneous machine translation (SMT), an output sequence should be produced as soon as possible, without reading the whole input sequence. This requirement creates a trade-off between translation delay and quality because less context may be known during translation. In most SMT methods, this trade-off is controlled with parameters whose values need to be tuned. In this paper, we introduce an SMT system that learns with reinforcement and is able to find the optimal delay in training. We conduct experiments on Tatoeba and IWSLT2014 datasets against state-of-the-art translation architectures. Our method achieves comparable results on the former dataset, with better results on long sentences and worse but comparable results on the latter dataset.

I. INTRODUCTION

SIMULTANEOUS machine translation (SMT) can be defined as producing output sequence tokens while reading input sequence tokens in an on-line fashion. These tokens may represent words in given languages, chunks of audio streams, or any other sequential data. The main difference between SMT and more general neural machine translation (NMT) is how the input and output sequences are processed. Most NMT methods read all input tokens and then generate the output sequence. Because of this, even though efficient state-of-the-art NMT methods exist, they cannot be used in SMT applications. Also, SMT methods need to consider the trade-off between delay and quality, as faster translation implies less context from the input. In most cases, this trade-off has to be optimized by checking various parameter settings, which is resource- and time-consuming.

SMT can be decomposed into a sequence of readings of the input tokens and writings of the output tokens. Reinforcement learning (RL) [1] is often applied to train SMT systems that sequentially choose between these actions and/or choose the written token. In this paper, we present an RL-based method with self-learning delay. Unlike in other approaches, we apply bootstrapping in training, which means that the sequences translated can be in principle infinite. We conduct experiments on Tatoeba and IWSLT2014 datasets against state-of-the-art translation architectures. Our method achieves comparable results on the former dataset, with better results on long sentences and worse but comparable results on the latter dataset.

The paper is organized as follows. Section II overviews literature related to neural machine translation, reinforcement learning, and simultaneous machine translation. Section III formally defines the problem considered in this paper. Section IV presents our method. Section V describes simulations

evaluating the presented architecture. Section VI discusses the experimental results and limitations of our approach. Section VII concludes the paper.

II. RELATED WORK

a) Neural machine translation (NMT): A basic architecture for neural machine translation includes an encoder that is fed with the input sequence; its final state becomes the initial state of a decoder that produces the output sequence [2]. In order to produce the right output, attention must be paid to significant input tokens. Attention was introduced to the encoder-decoder architecture in [3] and [4]. An architecture for NMT that is based solely on attention is Transformer [5]. Recurrent neural networks (RNN) were applied to capture short-term dependencies in input sequences and combined with multilayer attention in R-Transformer [6]. However, all these architectures only produce output when given the whole input sequence and hence are not applicable to on-line translation.

b) Reinforcement learning (RL): RL is a general framework for adaptation in the context of sequential decision making under uncertainty [1]. In this framework, an agent operates in discrete time, at each instant observing the state of its environment and taking action. Subsequently, the environment state changes, and the agent receives a numeric reward. Both the next state and the reward result from the previous state and action. By repeatedly facing the sequential decision problem in the same environment, the agent learns to designate actions in current environment states to be able to expect the highest future rewards.

In the context of this paper, especially interesting is the case where the agent cannot observe its state but only the value of a certain function of the state. This case is modeled as the Partially Observable Markov Decision Process (POMDP) [7]. In this model, the agent needs to collect subsequent observations to be able to recognize its current situation at any specific time. This can be done effectively with an RNN. Deep Recurrent Q-Learning [8] is an RL method for POMDP, which applies an RNN for that purpose.

RL has been applied to neural machine translation to optimize a policy, which, given an input sentence, assigned maximum probability to the corresponding output sentence. RL was applied this way to optimize the translation quality expressed in BLEU [9] which is not directly differentiable. RL has also been applied to train a random generator of sentences in

a generative adversarial architecture [10]. A similar architecture has been applied for the sequential generation of graphs [11].

c) *Simultaneous Machine Translation (SMT)*: A number of SMT methods use reinforcement learning. One of the first examples of using RL for SMT was presented in [12]. It uses imitation learning from the optimal sequence of actions to learn a policy for the system. In [13], a two-action framework was introduced, where the agent can read an input token, named READ, or write a new output token, named WRITE. This framework serves as a baseline for many new SMT methods, with authors extending and modifying it to achieve better results. The proposed reward function is based on the achieved BLEU score [14] and the translation delay metrics proposed by the authors, with the trade-off between delay and translation quality controlled by setting appropriate parameter values. In [15], a third action was added, named PREDICT, which works similarly to READ, but instead of reading an input token, it predicts this token. The reward function was also changed to include predictions' quality, with delay-quality trade-off still controlled by parameters. In [16], a commonly used NMT encoder-decoder structure was modified to work with SMT by making encoder, and attention dynamically change after every READ and adding an incremental decoder, which outputs a token from them after every WRITE. In [17], a method was proposed for extracting action sequences from NMT architectures, which were later used with sentence pairs in imitation learning to learn an optimal policy. Recently, reinforcement learning was used in multimodal translation [18], utilizing text and visual data to improve the quality of translations.

Not every SMT method uses reinforcement learning. In [19], the "wait- k " strategy was proposed, which produces a new output token with a fixed delay equal to k . It can be easily implemented in commonly used NMT architectures, shown by modifying the original Transformer. In [20], the "wait- k " strategy was used in speech-to-text task, showing it is efficient in applications other than machine translation.

III. PROBLEM DEFINITION

We consider input sequences, $x = (x_i)_{i=0}^{|x|-1}$, that contain tokens, $x_i \in \mathbb{R}^d$, $d \in \mathbb{N}$. The input sequences correspond to target sequences, $y = (y_j)_{j=0}^{|y|-1}$, $y_j \in \mathbb{R}^{d'}$, $d' \in \mathbb{N}$. The sequences are of variable lengths presented by the $|\cdot|$ function. An *interpreter agent* is fed with subsequent tokens from x and produces tokens of an output sequence, $(z_j)_{j=0}^{|z|-1}$, $z_j \in \mathbb{R}^{d'}$ on the basis of x .

Three special tokens playing various roles exist in both the input and the output space. They are:

- NULL — a missing element,
- EOS — denotes the last element of each sequence,
- PAD — an element concatenated to sequences after EOS for technical reasons.

For brevity, we will assume $x_i = \text{PAD}$, $y_j = \text{PAD}$, $z_j = \text{PAD}$ for, respectively, $i \geq |x|$, $j \geq |y|$, $j \geq |z|$.

Given x , the agent should produce z that minimizes the quality index in the form

$$J(y, z) = \sum_{j=0}^{K-1} L(y_j, z_j). \quad (1)$$

The loss L penalizes mistranslation; $L(\text{PAD}, \text{PAD}) = 0$; K is a number larger than any $|y|$. The sequence z that minimizes (1) is of length $|y|$, contains tokens equal to those in y , and ends with EOS.

We also require the interpreter agent to be of limited capacity but handle sequences of arbitrarily large lengths. In other words, we require the agent to operate on-line, i.e., it is fed with subsequent tokens of the input sequence and simultaneously produces subsequent tokens of the output sequence.

IV. METHOD

A. Reinforcement learning to transform sequences

We formalize the transformation of one sequence into another as an iterative decision process. At each of its instants, an agent reads a subsequent token from the input sequence or writes a subsequent token of the output sequence, similarly to [13]. That is, at each instant, the agent executes one of two actions:

- READ — another input token is read. This action is useful when it is (still) unclear what output token should be produced.
- WRITE — a subsequent output token is produced. This action is useful when a certain comprehensive portion of input tokens have been read, and a subsequent part of its interpretation can be presented.

A *policy* is a method of selecting actions and producing output tokens based on tokens read and those produced so far.

After execution of some of the actions, the agent receives numerical *rewards*. Let the rewards received during the process be denoted by $r = (r_k)_{k=0}^{|r|-1}$. A reward, r_k , is emitted at the following times:

- An output token, z_j , has just been written. Then r_k is the negative cost of mistranslation, i.e.

$$r_k = -L(y_j, z_j). \quad (2)$$

- A whole input sequence has been read, and the READ action is taken. This action does not make sense at this time. Therefore, for a certain constant $M > 0$, we have

$$r_k = -M. \quad (3)$$

Let $n(t)$ be the number of rewards emitted before the t -th action. The quality criterion for the policy is maximization of future discounted rewards. That is, at each time t the expected value of the *return*

$$R_t = \sum_{k=n(t)}^{|r|-1} \gamma^{k-n(t)} r_k \quad (4)$$

should be maximized, where $\gamma \in (0, 1)$ is a discount factor. In one episode of its operation, the agent transforms a single

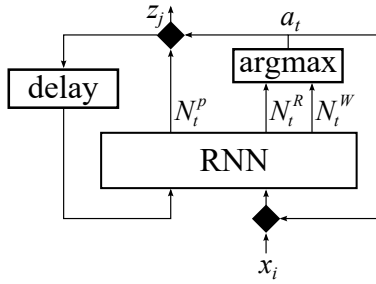


Fig. 1: Proposed architecture for on-line sequence transformation. The black squares represent passing/delaying x_i and outputting/skipping N_t^p depending on the action a_t .

sequence. It stops producing additional output tokens when it has outputted EOS.

In training, the agent, not having learned how to finish sequences, must be prevented from producing them infinitely long. Here we assume that an episode of training is terminated when the agent has produced as many tokens as in the target sequence y . The last target token is EOS, which is enough for the agent to learn to finish the output sequences.

Usually, in reinforcement learning [1] a reward comes after each action. However, here we want the agent to be rewarded only for the tokens it produces, bearing in mind that it does not produce them with READ actions. Rewards equal to zero for such actions do not make sense here because they could encourage the agent to maximize the sum of discounted rewards by postponing the production of output. Therefore, here we admit actions that are not immediately followed by rewards. Those emitted rewards have their own indices and are discounted according to them.

At each instant of its operation, a *state* of the agent's environment consists of the tokens the agent has read so far and the tokens it has written so far. However, before taking another action, it is only fed with the next input token and with the last written token. Therefore, the agent's environment is partially observable.

B. Architecture

We propose an architecture that learns to make the actions discussed above. The policy has the form of a recurrent neural network. Its input size is $d + d'$. In an instant of its operation it is fed with a subsequent input token concatenated with a preceding output token. Specifically, the first input to the network is the pair (x_0, NULL) . Let us assume that the agent has already read i input tokens and produced j output tokens. Thus, after the READ action, the network input is (x_i, NULL) . After the WRITE action, the network input is (NULL, z_{j-1}) .

In training *teacher forcing* can also be applied: The agent is fed not with the tokens it has already outputted but with target tokens.

Output of the network is of size $d' + 2$. The network produces a d' -dimensional *potential output token* and 2 scalar *return*

estimates that approximate returns (4) expected if actions WRITE and READ, respectively, were taken.

Let N_t be the $d' + 2$ -dimensional output of the network at t -th instant. It is composed as

$$N_t = [N_t^p, N_t^W, N_t^R],$$

where $N_t^p \in \mathbb{R}^{d'}$ is the potential output token, and $N_t^W, N_t^R \in \mathbb{R}$ are the return estimates for the WRITE and READ actions, respectively. The architecture is depicted in Fig. 1.

The network output that estimates the return corresponding to the just taken action a_t is trained to approximate the conditional expected value

$$Q_t(a_t) = E(R_t | C_t), \quad (5)$$

where the condition C_t includes the following:

- 1) The action just taken is a_t .
- 2) Subsequently, those actions are selected, which correspond to the network return estimates with maximum values.
- 3) Input tokens read so far, and output tokens produced so far. At the time t , the rest of the tokens are unknown, thereby remaining random vectors.

The actions actually taken are usually selected as those maximizing the return estimates given by the network. However, with a small probability, the agent chooses the other action since it needs to explore different actions to learn their consequences. Therefore, we will not estimate $Q_t(a_t)$ based on the actual return, but on a recursion instead. Specifically, let us denote by j the number of output tokens produced before the analyzed action is taken. A simple analysis reveals that $Q_t(a_t)$ (5) satisfies the following recursive equation:

$$Q_t(a_t) = E \left\{ \begin{array}{l} -M + \gamma \max_b Q_{t+1}(b) \\ \quad \text{if } a_t = \text{READ}, \text{fin}(x) \\ \max_b Q_{t+1}(b) \\ \quad \text{if } a_t = \text{READ}, \neg \text{fin}(x) \\ -L(y_j, z_j) + \gamma \max_b Q_{t+1}(b) \\ \quad \text{if } a_t = \text{WRITE}, j < |y| - 1 \\ -L(y_j, z_j) \\ \quad \text{if } a_t = \text{WRITE}, j = |y| - 1 \end{array} \middle| a_t \right\} \quad (6)$$

where $\text{fin}(x)$ means that all x tokens have been read. The condition for the above expectation is that the action actually taken is a_t .

Target values for the network will be based on the above recursive equation and the fact that $Q_{t+1}(\text{READ})$ and $Q_{t+1}(\text{WRITE})$ are estimated by N_{t+1}^R and N_{t+1}^W , respectively. Therefore, the network outputs at time t are trained as follows.¹ After the READ action, when x is not finished yet, N_t^R is adjusted:

$$N_t^R \leftarrow \max\{N_{t+1}^R, N_{t+1}^W\}. \quad (7)$$

¹We apply the notation:

$$[\text{predicate}] = \begin{cases} 1 & \text{if } \text{predicate} \text{ is true} \\ 0 & \text{otherwise.} \end{cases}$$

After the READ action, when x is already finished:

$$N_t^R \leftarrow -M + \gamma \max\{N_{t+1}^R, N_{t+1}^W\}. \quad (8)$$

After a WRITE action, the return estimate for the WRITE action is adjusted as

$$N_t^W \leftarrow -L(N^p, z_j) + [j < |y| - 1] \gamma \max\{N_{t+1}^R, N_{t+1}^W\}, \quad (9)$$

Also, the potential output is adjusted

$$N_t^p \leftarrow y_j. \quad (10)$$

C. Weighting losses due to mistranslation and return estimation

The network produces outputs of two qualitatively different kinds: the potential output tokens and the return estimates. The network training requires minimization of an aggregated loss that combines a loss due to mistranslation and a loss due to return estimation. We propose to normalize these losses with their averages defined below.

Let $n = 1, 2, \dots$ be a training minibatch index. We average original mistranslation losses, L_n^M , and original estimation losses, L_n^E , according to

$$\bar{L}_n^M = w_n \bar{L}_{n-1}^M + (1 - w_n) L_n^M, \quad (11)$$

$$\bar{L}_n^E = w_n \bar{L}_{n-1}^E + (1 - w_n) L_n^E \quad (12)$$

where $\bar{L}_0^M = \bar{L}_0^E = 0$, and

$$w_n = \rho(1 - \rho^{n-1}) / (1 - \rho^n), \quad (13)$$

where $\rho \in (0, 1)$ is the decay factor, e.g. $\rho = 0.99$. The terms (11,12) approximate arithmetic means for small n , and exponential moving average for larger n .

Training the network aims at minimizing the aggregated loss in the form

$$L_n = L_n^M / \bar{L}_n^M + \eta(n, n_0) L_n^E / \bar{L}_n^E. \quad (14)$$

The term $\eta(n, n_0)$ is a relative weight of the estimation loss for the current minibatch/epoch index n and the (expected) total number n_0 of minibatches/epochs in the whole training. For small n this weight should be small: $\eta(n, n_0) \approx \eta_{min}$, since high accuracy of future rewards is pointless when quality of outputted tokens is poor. $\eta(n, n_0)$ is gradually growing with n to a certain asymptote, η_{max} . η_{min} and η_{max} are hyperparameters of the training process, e.g. 1/50 and 1/5, respectively. The η function may have the form

$$\eta(n, n_0) = \eta_{max} - (\eta_{max} - \eta_{min}) \exp(-3n/n_0). \quad (15)$$

V. EXPERIMENTAL STUDY

In this section, we demonstrate the effectiveness of our proposed architecture, henceforth called RLST (Reinforcement Learning for on-line Sequence Transformation). We perform experiments with seven machine translation tasks. They are based on datasets taken from Tatoeba [21] and dataset taken from IWSLT2014 [22].

In our machine translation tasks, the input sequence consists of tokens representing words of a sentence in a source language. The aim is to generate a sequence of tokens with

the same sentence meaning as the source sequence. We conduct experiments on datasets presented in Table I which contains basic statistics on the source and target languages datasets, sizes of source and target dictionaries, and numbers of sentences in each data split. For Tatoeba datasets, we also separate long test splits, where source sentences have more than 22 tokens. The long test split allows us to compare how models deal with longer input sentences. For all datasets, we compare our proposed RLST architecture with state-of-the-art machine translation architectures, namely encoder-decoder with attention [3] and Transformer [5]. For both encoder-decoder and Transformer, the minimized loss is cross-entropy. For RLST, we quantify L_n^M and L_n^E in (14) as cross-entropy and mean square error, respectively.

In our experiments, we employed the following procedure to optimize hyperparameters of the compared architectures. For Tatoeba datasets, we optimized the hyperparameters manually for all three architectures to obtain their best BLEU score [14] on the En-Es language pair and applied these values to all language pairs. For IWSLT2014 datasets, we optimized the hyperparameters of RLST manually and took the hyperparameters for the Transformer from [5] and for the encoder-decoder with attention from [23].

Our simulation experiments have been performed on a PC equipped with AMD Ryzen™ Threadripper™ 1920X, 64GB RAM, 4xNVIDIA™ GeForce™ RTX 2070 Super™.

A. Tatoeba

Tatoeba datasets [21] contain various, mostly unrelated, sentences and their translations provided by the community. We preprocess them using spaCy tokenizer [24] and replace tokens that appear in training corpora less than three times with a unique token representing an unknown word. We also remove duplicated source sentences.

Experiments on Tatoeba for all architectures are run for 50 epochs, with a batch size of 128 and gradient clipping norm set to 10.0. Encoder-decoder and RLST have weight decay set to 10^{-5} , while Transformer has weight decay set to 10^{-4} . Source and target tokens are converted to trainable vectors of length 256 initialized with $\mathcal{N}(0, 1)$. There is a dropout applied to them with a probability of 0.2. We use Adam optimizer with default parameters and a constant learning rate equal to 0.0003. The reference encoder-decoder is presented in [3]. Its encoder is a bidirectional GRU recurrent layer with 256 hidden neurons followed by a linear attention layer with 64 neurons. The decoder is a GRU recurrent layer with 256 hidden neurons followed by a dropout with 0.5 probability and a linear output layer with a number of neurons equal to the target's vocabulary size. The teacher forcing ratio for encoder-decoder during training is set to 1.0. For the Transformer, we use the following parameters: the number of expected features in the encoder and decoder inputs is 256, the number of heads in multiattention is 8, the number of encoder and decoder layers is 6, the dimension of feedforward layers is 512, the dropout probability is 0.25 and the teacher forcing ratio to 1.0. For RLST, we use the following approximator. Input and

Dataset	Abbr	Src. dict.	Trg. dict.	Train set	Valid. set	Test set	Long test
Tatoeba Spanish-English	Tat Es-En	13 288	8 960	124 179	41 393	41 394	2 387
Tatoeba French-English	Tat Fr-En	13 792	10 056	161 283	53 761	53 762	2 613
Tatoeba English-Spanish	Tat En-Es	8 690	12 698	115 026	38 342	38 342	2 325
Tatoeba English-Russian	Tat En-Ru	10 009	21 820	241 785	80 595	80 595	1 756
Tatoeba English-German	Tat En-De	10 504	15 276	170 347	56 782	56 783	3 805
IWSLT2014-German-English	IWSLT-De-En	8 848	6 632	160 239	7 283	6 750	—
IWSLT2014-English-German	IWSLT-En-De	6 632	8 848	160 239	7 283	6 750	—

TABLE I: Basic statistics of machine translation datasets.

previous output embeddings with a dimension of 256 are passed to a dense layer with 512 neurons, Leaky ReLU activation with negative slope set to 0.01 and dropout probability of 0.2. Its output is processed by four GRU layers with the hidden dimension of 512 and residual connections between them. The output of the last recurrent layer is passed to a dense layer with 512 neurons, Leaky ReLU activation with a negative slope set to 0.01, and a dropout probability of 0.5. The output of the last dense layer is passed to the output linear layer with number of neurons equal to the target’s vocabulary size and additional 2 neurons representing Q-values of actions. We also set $\gamma = 0.9$, $\varepsilon = 0.3$, $M = 3$, $N = 50000$, $\eta_{min} = 0.02$, $\eta_{max} = 0.2$, $\rho = 0.99$ and teacher forcing ratio to 1.0.

B. IWSLT2014

We conduct experiments on IWSLT2014 German-English and English-German datasets using the *fairseq* framework [22]. Data is preprocessed using the script provided by the benchmark, which utilizes byte-pair encoding (BPE) [25]. For every architecture, the training lasts for 100 epochs with varying batch sizes to ensure that the maximum number of tokens in a batch equals 4096 and the gradient clipping norm is set to 10.0. The encoder-decoder and RLST architectures are trained using Adam optimizer with default parameters and constant learning rate scheduling with weight decay of 10^{-5} . The Transformer is also trained using Adam optimizer, with parameters and a learning rate scheduler described in [5]. For the encoder-decoder and the Transformer, we use the *lstm_wiseman_iwslt_de_en* architecture (based on [23]) and *transformer_iwslt_de_en* (based on [5] with some changes), respectively. The encoder-decoder model has trainable source and target embeddings dimensions of 256 without dropout. Its encoder is an LSTM layer with 256 hidden neurons, and its decoder was also an LSTM layer with 256 hidden neurons followed by an output layer with the number of neurons equal to the target’s vocabulary size. The decoder uses the attention mechanism. The encoder and decoder layers have a dropout probability of 0.1. The Transformer has the following parameters: The trainable source and target embeddings dimensions are 512 without dropout, the number of neurons in feedforward layers is 1024, the number of multiattention heads is 4, the number of encoder and decoder layers is 6 and a dropout probability of 0.1. For RLST, we set trainable source and target embedding dimensions to 256 with a dropout of 0.2 probability. In the case of IWSLT-En-De, we

use the same approximator as in Tatoeba. For IWSLT-De-En, we changed the dimensions of dense and GRU layers from 512 to 768. We also set $\gamma = 0.9$, $\varepsilon = 0.30$, $M = 7$, $N = 100000$, $\rho = 0.99$, $\eta_{min} = 0.02$, $\eta_{max} = 0.2$ and teacher forcing ratio to 1.0. For encoder-decoder and Transformer, we set beam search width to 1 and teacher forcing ratio to 1.0.

C. Results

The results are presented in Table II. For each dataset and architecture, we show BLEU values computed on a test split from checkpoints for which the BLEU value on a validation split was the highest. We also show the number of parameters for each model. The highest values of BLEU for each dataset are bolded. On the Tatoeba test confined to sentences of length up to 22 words, all three architectures achieved similar BLEU. However, in the test confined to longer sentences, RLST outperforms the other architectures in 4 language pairs out of 5, usually by a large margin. The architecture to achieve the best results on fairseq datasets is the Transformer. RLST and the encoder-decoder with attention achieve similar BLEU on this benchmark.

In order to gain an additional insight into the operation of the RLST interpreter agent we present in Figure 2 the timing of taking the READ and WRITE actions. As one may expect, initially, the agent is mostly reading, then reading and writing ratios are roughly equal, and finally, the agent is mostly writing. It appears that the agent has read about five more words than it has written for most of the time. That seems to correspond to a common intuition: A human interpreter also needs to be delayed a few words in producing an accurate translation of a speech.

VI. DISCUSSION

Our proposed interpreter agent RLST is designed to transform arbitrarily long sequences on-line. In each cycle of its operation, it performs the same number of computations in which it reads an input token or writes an output token. The agent has only limited memory space to store information about recently read tokens, a context of these tokens defined by previous ones, and recently written tokens. Therefore, the agent is not a method of choice for translating sentences of moderate length without any context.

The RLST architecture outperformed others in the test on long sentences (longer than 22 words) taken from Tatoeba. The memory state of the interpreter agent preserved the context

Architecture → Dataset ↓	Encoder-decoder		Transformer		RLST	
	BLEU	Num. params	BLEU	Num. params	BLEU	Num. params
Tat Es-En	50.33	16 637 504	50.19	15 906 560	50.02	17 122 050
Tat Es-En (L)	16.52	16 637 504	13.42	15 906 560	20.57	17 122 050
Tat Fr-En	53.95	18 170 504	53.89	16 597 832	53.05	18 093 898
Tat Fr-En (L)	13.49	18 170 504	10.03	16 597 832	16.42	18 093 898
Tat En-Es	45.14	20 248 794	44.63	16 647 066	45.09	18 819 484
Tat En-Es (L)	16.1	20 248 794	12.39	16 647 066	21.07	18 819 484
Tat En-Ru	47.71	32 271 740	47.11	21 664 316	47.37	26 171 966
Tat En-Ru (L)	10.06	32 271 740	5.66	21 664 316	11.28	26 171 966
Tat En-De	41.98	24 015 596	41.63	18 433 964	40.62	21 266 350
Tat En-De (L)	10.95	24 015 596	9.5	18 433 964	10.2	21 266 350
IWSLT De-En	24.13	7 178 728	32.17	42 864 640	23.28	24 223 210
IWSLT En-De	19.01	7 748 240	26.13	43 999 232	18.32	15 331 986

TABLE II: BLEU scores on test splits and number of parameters for tested architectures. (L) on Tatoeba datasets denotes scores from long test split.

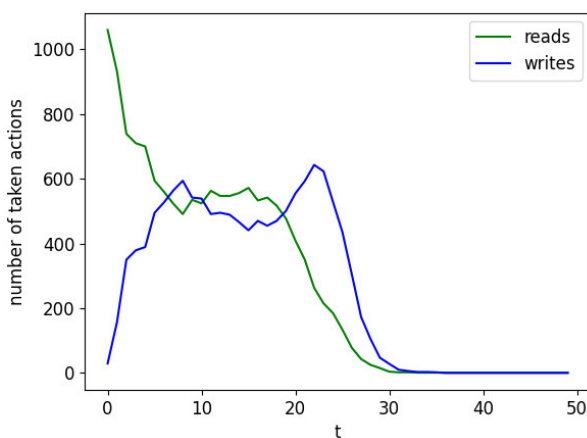


Fig. 2: Processing of 1089 source sentences of length 15 from the Tat En-Ru test dataset by the RLST interpreter agent. The graph shows when the READ and WRITE actions are taken.

of the outputted words better than the attention mechanism managed to do in the reference architectures. We hypothesize that the sequential nature of human language makes it possible to translate properly separate parts of a speech, but in order to do that, the end of each part must be identified. It appears that RLST manages to do it better in long sentences than the reference architectures.

The goal of a large fraction of algorithms developed in computer science is to transform input data into output data whose size is unknown in advance. For some data types, it is natural to process them sequentially. These types include natural language, sound, video, and bioinformatic data, e.g., genetic. The experiments in Section V confirm that our introduced RLST architecture is very well adapted to such data.

VII. CONCLUSIONS

In this paper, we have presented the RLST architecture that transforms on-line sequences of arbitrary length without the need to define the trade-off between delay and quality. In the transformation process, it makes sequential decisions about

whether to read an input token or write an output token. The architecture learns to make these decisions with reinforcement. The experimental study compared the architecture with state-of-the-art machine translation methods, namely the Transformer and the encoder-decoder with attention. Benchmark datasets taken from Tatoeba and IWSLT with seven language pairs were employed in the experiments. The RLST architecture solved a more complex problem of on-line transformation than the reference methods, which produced output tokens knowing the entire source sequence. Even so, RLST produced translations of comparable quality. It also outperformed reference architectures in tests with long sentences (longer than 22 words) taken from Tatoeba. That confirms that it is particularly well suited to applications in which transformation of sequences of arbitrary lengths and/or on-line is required.

ACKNOWLEDGMENTS

The project was funded by POB Research Centre for Artificial Intelligence and Robotics of Warsaw University of Technology within the Excellence Initiative Program – Research University (ID-UB).

REFERENCES

- [1] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. The MIT Press, 2018. doi: <https://dx.doi.org/10.1109/TNN.1998.712192>
- [2] I. Sutskever, O. Vinyals, and Q. V. Le, “Sequence to sequence learning with neural networks,” 2014, arXiv:1409.3215. doi: <https://dx.doi.org/10.48550/arXiv.1409.3215>
- [3] D. Bahdanau, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” 2015. doi: <https://dx.doi.org/10.48550/arXiv.1409.0473>
- [4] T. Luong, H. Pham, and C. D. Manning, “Effective approaches to attention-based neural machine translation,” in *Conference on Empirical Methods in Natural Language Processing*, 2015. doi: <https://dx.doi.org/10.18653/v1/D15-1166> pp. 1412–1421.
- [5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” in *NIPS*, 2017.

- [6] Z. Wang, Y. Ma, Z. Liu, and J. Tang, “R-transformer: Recurrent neural network enhanced transformer,” 2019, arXiv:1907.05572. doi: <https://dx.doi.org/10.48550/arXiv.1907.05572>
- [7] S. Kapturowski, G. Ostrovski, J. Quan, R. Munos, and W. Dabney, “Recurrent experience replay in distributed reinforcement learning,” in *International Conference on Learning Representations*, 2019.
- [8] M. Hausknecht and P. Stone, “Deep recurrent q-learning for partially observable mdps,” 2015, arXiv:1507.06527. doi: <https://dx.doi.org/10.48550/arXiv.1507.06527>
- [9] L. Wu, F. Tian, T. Qin, J. Lai, and T.-Y. Liu, “A study of reinforcement learning for neural machine translation,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018. doi: <https://dx.doi.org/10.18653/v1/D18-1397> pp. 3612–3621.
- [10] L. Yu, W. Zhang, J. Wang, and Y. Yu, “Seqgan: Sequence generative adversarial nets with policy gradient,” in *AAAI*, 2017.
- [11] G. L. Guimaraes, B. Sanchez-Lengeling, P. L. C. Farias, and A. Aspuru-Guzik, “Objective-reinforced generative adversarial networks (organ) for sequence generation models,” 2017, arXiv:1705.10843. doi: <https://dx.doi.org/10.48550/arXiv.1705.10843>
- [12] A. Grissom II, H. He, J. Boyd-Graber, J. Morgan, and H. Daumé III, “Don’t until the final verb wait: Reinforcement learning for simultaneous machine translation,” in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014. doi: <https://dx.doi.org/10.3115/v1/D14-1140> pp. 1342–1352.
- [13] J. Gu, G. Neubig, K. Cho, and V. O. Li, “Learning to translate in real-time with neural machine translation,” in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, 2017. doi: <https://dx.doi.org/10.18653/v1/E17-1099> pp. 1053–1062.
- [14] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, “Bleu: a method for automatic evaluation of machine translation,” in *40th annual meeting on association for computational linguistics*, 2002. doi: <https://dx.doi.org/10.3115/1073083.1073135> pp. 311–318.
- [15] A. Alinejad, M. Siahbani, and A. Sarkar, “Prediction improves simultaneous neural machine translation,” in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium, 2018. doi: <https://dx.doi.org/10.18653/v1/D18-1337> pp. 3022–3027.
- [16] F. Dalvi, N. Durrani, H. Sajjad, and S. Vogel, “Incremental decoding and training methods for simultaneous translation in neural machine translation,” in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 2018. doi: <https://dx.doi.org/10.18653/v1/N18-2079> pp. 493–499.
- [17] B. Zheng, R. Zheng, M. Ma, and L. Huang, “Simpler and faster learning of adaptive policies for simultaneous translation,” in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019. doi: <https://dx.doi.org/10.18653/v1/D19-1137> pp. 1349–1354.
- [18] J. Ive, A. M. Li, Y. Miao, O. Caglayan, P. Madhyastha, and L. Specia, “Exploiting multimodal reinforcement learning for simultaneous machine translation,” in *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, 2021. doi: <https://dx.doi.org/10.18653/v1/2021.eacl-main.281> pp. 3222–3233.
- [19] M. Ma, L. Huang, H. Xiong, R. Zheng, K. Liu, B. Zheng, C. Zhang, Z. He, H. Liu, X. Li, H. Wu, and H. Wang, “STACL: Simultaneous translation with implicit anticipation and controllable latency using prefix-to-prefix framework,” in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019. doi: <https://dx.doi.org/10.18653/v1/P19-1289> pp. 3025–3036.
- [20] Y. Ren, J. Liu, X. Tan, C. Zhang, T. Qin, Z. Zhao, and T.-Y. Liu, “SimulSpeech: End-to-end simultaneous speech to text translation,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020. doi: <https://dx.doi.org/10.18653/v1/2020.acl-main.350> pp. 3787–3796.
- [21] Tatoeba, “<https://tatoeba.org>,” 2020, retrieved 2020-05-05.
- [22] M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, and M. Auli, “fairseq: A fast, extensible toolkit for sequence modeling,” in *Proceedings of NAACL-HLT 2019: Demonstrations*, 2019. doi: <https://dx.doi.org/10.18653/v1/N19-4009>
- [23] S. Wiseman and A. M. Rush, “Sequence-to-sequence learning as beam-search optimization,” *arXiv preprint arXiv:1606.02960*, 2016. doi: <https://dx.doi.org/10.48550/arXiv.1606.02960>
- [24] M. Honnibal, I. Montani, S. Van Landeghem, and A. Boyd, “spaCy: Industrial-strength Natural Language Processing in Python,” 2020. doi: <https://dx.doi.org/10.5281/zenodo.1212303>
- [25] R. Sennrich, B. Haddow, and A. Birch, “Neural machine translation of rare words with subword units,” *arXiv preprint arXiv:1508.07909*, 2015. doi: <https://dx.doi.org/10.48550/arXiv.1508.07909>

Applying SoftTriple Loss for Supervised Language Model Fine Tuning

Witold Sosnowski

Faculty of Mathematics
and Information Science

Warsaw University of Technology
Warsaw, Poland

Email: witold.sosnowski.dokt@pw.edu.pl
ORCID: 0000-0002-2241-9588

Anna Wróblewska

Faculty of Mathematics
and Information Science

Warsaw University of Technology
Warsaw, Poland

Email: anna.wroblewska1@pw.edu.pl
ORCID: 0000-0002-3407-7570

Piotr Gawrysiak

Faculty of Electronics
and Information Technology

Warsaw University of Technology
Warsaw, Poland

Email: p.gawrysiak@ii.pw.edu.pl
ORCID: 0000-0002-9647-6761

Abstract—We introduce a new loss function based on cross entropy and SoftTriple loss, TripleEntropy, to improve classification performance for fine-tuning general knowledge pre-trained language models. This loss function can improve the robust RoBERTa baseline model fine-tuned with cross-entropy loss by about 0.02–2.29 percentage points. Thorough tests on popular datasets using our loss function indicate a steady gain. The fewer samples in the training dataset, the higher gain—thus, for small-sized dataset, it is about 0.71 percentage points, for medium-sized—0.86 percentage points, for large—0.20 percentage points, and for extra-large 0.04 percentage points.

I. INTRODUCTION

NATURAL language processing (NLP) is a rapidly growing area of machine learning with applications wherever a computer needs to operate on a text that involves capturing its semantics. It may include text classification, translation, text summarization, question answering, and dialogues. All these tasks are downstream and depend on the quality of the text representation [1]. Many models can produce such text representations, from Bag-of-Word (BoW) or Word2Vec word embedding to the state-of-the-art language representation model BERT with variations in most NLP tasks.

The best performance on text classification tasks is obtained when the model is first trained on a general knowledge corpus to capture semantic relationships between words and then fine-tuned with an additional dense layer on a domain corpus with cross-entropy loss [2].

We introduce a new loss function – TripleEntropy – to improve classification performance for fine-tuning general knowledge pre-trained language models based on cross-entropy loss and SoftTriple loss [3], [4]. Triplet Loss transforms the embedding space so that vector representations from the same class can form separable subspaces, stabilizing and generalizing the language model fine-tuning process. TripleEntropy can improve the fine-tuning process of the RoBERTa based models, so the performance on downstream tasks increases by about 0.02 - 2.29 percentage points.

In the following sections, we review relevant work on state-of-the-art in distance metric learning (Section II); describe our approach for training and our metric SoftTriple loss and outline the experimental setup (Section III); discuss the results

(Section IV); conclude and offer directions for further research (Section V).

II. RELATED WORK

A. Building Sentence Embeddings

Building embeddings that represent sentences is challenging because the natural language can be very diverse. The meaning can change drastically depending on the context of a word. It is also an important issue because the quality of sentence embeddings substantially impacts the performance of all downstream tasks like text classification and question answering. Because of that, so far, considerable research effort has been put into building sentence embeddings.

One of the first vector representations (embeddings), BoW, is an intriguing approach in which the text is represented as a bag (multiset) of its words, with each word represented by its occurrence in the text [5]. The disadvantage of this strategy was that the BoW embeddings fail to capture hidden meaning of words and sentences, unlike the Word2Vec approach, which used a machine learning process to predict word embeddings [6] and is able to represent the latent meaning of the word. In Word2Vec, each word embedding is selected based on its overall context in the training corpus and can express the latent semantics of words. Unfortunately, this method does not express the semantics of the whole sentence, so several approaches have been proposed to solve this problem. The most popular approaches build the sentence embedding as a weighted average of the sentence's word vectors. Since in Word2Vec every word embedding is static, regardless of its meaning in the whole sentence, this approach is not adapted to changes in sentence and context semantics.

Bidirectional Encoder Representations from Transformers (BERT) is a well-known technique for constructing high-quality sentence embeddings that can express the dynamic and latent meaning of the whole sentences better than any previous approach. Its sentence embeddings can accurately reflect the meaning of the input text, making a significant difference in the quality of the downstream tasks performed. An even better variant of the BERT-based architecture, RoBERTa, has

emerged and has lately become unquestionably state-of-the-art in terms of sentence embedding construction [7], [8].

B. Distance Metric Learning

Embedding learning that exploits the fact that instances from the same class are closer than instances from other classes is known as Distance Metric Learning (DML) [4]. DML recently has drawn much attention due to its wide applications, especially in image processing. It can be used in the classification tasks together with the k-nearest neighbour algorithm [9], clustering along with K-means algorithm [10] and semi-supervised learning [11]. DML's objective is to create embeddings similar to examples from the same class but different from observations from other classes. [12]. In contrast to the cross-entropy loss, which only takes care of intra-class distances to make them linearly separable, the DML approach maximizes inter-class and minimizes the intra-class distances [13]. Aside from that, a typical classifier based solely on cross-entropy loss concentrates on class-specific characteristics rather than generic features of the dataset, as it is only concerned with distinguishing between classes rather than learning their representations. DML focuses on learning class representations, making the model more generalizable to new observations and more robust to outliers. There are various DML methods in use today, of which the following are the most important.

1) *Contrastive Loss*: Contrastive Loss (CL) is one of the earliest methods in DML [14]. It concentrates on pairs of similar and dissimilar observations¹, whose distances are attempted to be minimized if they belong to the same class and maximized if they belong to different classes. The CL method is given in Equation 1.

$$\ell = \sum_{i=1}^N \left[(1 - y_i) \frac{1}{2} (d(z_i^1, z_i^2))^2 + (y_i) \frac{1}{2} \left\{ \max(0, m - d(z_i^1, z_i^2)) \right\}^2 \right] \quad (1)$$

where i denotes the index of a pair of representations z_i^1, z_i^2 of the sample pair x_i^1, x_i^2 from the set of all pairs \mathcal{P} in the training set with the cardinality N . y_i denotes label assigned to the i th pair. It has a value of 0 if the associated samples x_i^1 and x_i^2 belong to the same class, otherwise it has a value of 1. $d()$ is the Euclidean distance functions between a pair of representations z_i^1, z_i^2 . $m > 0$ is the margin beyond which dissimilar points have no effect on the loss.

2) *Triplet Loss*: Triplet Loss, as another solution to the DML problem, is similar to Contrastive Loss but works with triplets instead of pairs [15]. Each triplet comprises an anchor, a positive, and a negative observation. Positive examples are members of the same class as an anchor, but negative instances belong to a separate class. Because it considers more observations simultaneously, it optimizes the embedding space

¹In this paper we use: observations, samples, examples as synonyms. We even refer to sentences as observations because most datasets contain one sentence as an observation, i.e., the input to the ML model

better than Contrastive Loss. The actual formula for Triplet Loss is in Equation 2.

$$\ell = \sum_{i=1}^N \left[\|z_i^a - z_i^p\|_2^2 - \|z_i^a - z_i^n\|_2^2 + m \right]_+ \quad (2)$$

where i is the index of a triplet of representations z_i^a, z_i^p, z_i^n of the samples x_i^a, x_i^p, x_i^n from the set of all triplets \mathcal{T} in the training set with the cardinality N . x_i^a denotes an *anchor*, x_i^p (*positive*) is the observation from the same class as the anchor, x_i^n (*negative*) denotes an observation belonging to a different than the anchor class, m is a margin imposed between positive and negative pairs margin.

The most typical issue with triplets and contrastive learning is that as the number of observations in a batch grows, the number of pairs and triplets grows squarely or cubically. Another point is that using training pairs and triples, which are relatively easy to distinguish, leads to poor generalization of the model. Semi-solutions of the above problems are as introducing τ a temperature parameter that controls the separation of classes [16], or hard triples, which samples such triplets that the anchor and the positive are not close together, and the anchor and the negative are close together [17].

Triplet Loss has previously been used with the BERT language model to detect whether new claims are similar to a set of claims that were previously fast-checked online [18]. Another interesting work uses self-supervised triplet training to learn similarities for recommendations [19]. The triplet network was also used with the BERT encoder in the domain of protein modelling to solve several regression tasks with limited data such as peak absorption wavelength or enantioselectivity [20].

3) *ProxyNCA Loss*: It is a more general approach to solving a problem with high resource consumption [12]. It employs proxies – artificial data points in the representation space that represent the entire dataset. One proxy approximates one class; therefore, there are as many proxies as classes. This technique drastically reduces the number of triplets while simultaneously raising the convergence rate since each proxy makes the triplet more resistant to outliers. The proxies are integrated into the model as trainable parameters since the synthetic data points are represented as embeddings. Equation 3 depicts a ProxyNCA loss formula.

$$\ell = -\frac{1}{N} \sum_{i=1}^N \log \left(\frac{\exp(-d(z_i, p(y_i)))}{\sum_{p(ne) \in p(Ne_i)} \exp(-d(z_i, p(ne)))} \right) \quad (3)$$

where i denotes the index of the representation z_i of the observation x_i , N indicates the number of observations in the training set, $p(y_i)$ denotes the anchor's proxy, $p(Ne_i)$ denotes proxies representing the different classes that x_i belongs, d is the Euclidean distance between the anchor and the given proxy.

4) *SoftTriple Loss*: A single proxy per class may not be enough to represent the class's inherent structure in real-world data. Another DML loss function introduces multiple

proxies per class - SoftTriple Loss [4]. It can produce better embeddings while maintaining a smaller number of triplets than Triplet Loss or Contrastive Loss. The SoftTriple Loss is defined by the Equations 4, 5 and 6.

$$\ell_{SoftTriple} = -\frac{1}{N} \sum_{i=1}^N \ell_{ST_i} \quad (4)$$

$$\ell_{ST_i} = -\log \frac{\exp(\lambda(S'_{i,y_i} - \delta))}{\exp(\lambda(S'_{i,y_i} - \delta)) + \sum_{j \neq y_i} \exp(\lambda S'_{i,j})} \quad (5)$$

$$S'_{i,c} = \sum_k \frac{\exp\left(\frac{1}{\gamma} z_i^\top w_c^k\right)}{\sum_k \exp\left(\frac{1}{\gamma} z_i^\top w_c^k\right)} z_i^\top w_c^k \quad (6)$$

where N denotes the number of observations in the training set, $c \in C$ indicates the class index, C indicates number of classes, k is the number of proxies per class, δ defines a margin between the example and class centres from different classes, λ reduces the influence from outliers and makes the loss more robust, γ is the scaling factor for the entropy regularizer, z_i defines the representation of the observation x_i , w_c^k denotes proxy embeddings of the class c (there are k of them).

III. OUR APPROACH

For fine-tuning pre-trained language models, we offer a novel objective function TripleEntropy. It is based on the supervised cross-entropy loss and the SoftTriple Loss [4]. The latter component is a loss from the Distance Metric Learning (DML) family of losses, which learns an embedding by capturing similarities between embeddings from the same class and distinguishing them from embeddings from different classes [4].

For the classification problem, let us denote (as in the previous section):

- N – the number of observations,
- $c \in C$ the class index, where C indicates the number of classes,
- y_{ic} – the objective probability of the class c for the i th observation,
- β – the scaling factor that tunes influence of both parts of the loss.

The TripleEntropy is given by the Equation 7:

$$\mathcal{L} = (\beta)\ell_{MCE} + (1 - \beta)\ell_{SoftTriple} \quad (7)$$

where

$$\ell_{MCE} = -\frac{1}{N} \sum_i \sum_c y_{ic} \log(p_{ic}) \quad (8)$$

It can be applied for different encoders $E(\cdot) \in \mathbf{R}^d$ from natural language processing domain such as BERT [3], RoBERTa [7] or others models that create text representations (embedding).

A. Model

In our work, we use the objective function from Equation 7 to fine-tune the pre-trained BERT-based language models provided by the *huggingface* library as RoBERTa-base and RoBERTa-large as depicted in the figure 1. In the standard settings, the single input text is first tokenized with Byte-Pair Encoding (BPE) tokenizer [2], which produces a vector of tokens x_i with a maximum length of 512, with $[CLS]$ at the beginning of an array, $[EOS]$ at the end and $[SEP]$ between tokens representing different sentences. The output of RoBERTa model $E(x_i) \in \mathbf{R}^d$ is an array of embeddings, where each input token has its corresponding embedding.

1) *Multinomial Cross-Entropy Loss*: In our experiments, we used the multinomial cross-entropy (MCE) loss calculated in the same way as it was proposed by the authors of the BERT language model [3]. The sentence representation is obtained by pooling the output of the model $E(x_i) \in \mathbf{R}^d$ and passing it to the C dimensional single fully connected layer. Its output is passed to the softmax function generating probabilities p_{ic} , which are, along with objective probabilities y_{ic} , directly feeding the multinomial cross-entropy loss.

2) *SoftTriple Loss*: The second component of the TripleEntropy loss Equation 7 is SoftTriple Loss Equation 4, responsible for a more robust and better generalization of the model during tuning. It is fed by the direct output of the model $E(x_i) \in \mathbf{R}^d$, even before pooling. It means that if the batch size is B , then the total number of embeddings that feed SoftTriple Loss during one training iteration is $B * |x_i|$. This implementation ensures that the proxies representing each class will be well approximated so that the quality of fine-tuning increases.

Our implementation is a development of the earlier work [21], where Contrastive Loss was applied only to the embedding corresponding to the first $[CLS]$ token of the input vector x_i . We apply SoftTriple Loss to the embeddings corresponding to all tokens from the input vector x_i , which ensures the better generalization of the fine-tuning process but requires more computing power. Fortunately, the SoftTriple Loss is significantly more efficient than the Contrastive Loss since it generates triplets not from all observations but from its approximated proxies.

B. Training and testing

During our experiments, each result (average accuracy) was obtained as based on 4 seed runs (2, 16, 128, 2048), where each run was 5-fold cross-validated. It means that each accuracy result is an averaged of 20 different results. Apart from that, each result was based on the best parameter combination obtained by grid search which included parameters $k \in \{10, 100, 1000, 2000\}$, $\gamma \in \{0.01, 0.03, 0.05, 0.07, 0.1\}$, $\lambda \in \{1, 3, 3.3, 4, 6, 8, 10\}$, $\delta \in \{0.01, 0.1, 0.3, 0.5, 0.7, 0.9, 1\}$ and $\beta \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$. We noticed that for most experiments, the best hyperparameter set is following $k = 2,000$, $\gamma = 0.1$, $\lambda = 3.3$, $\delta = 0.3$ and $\beta = 0.9$.

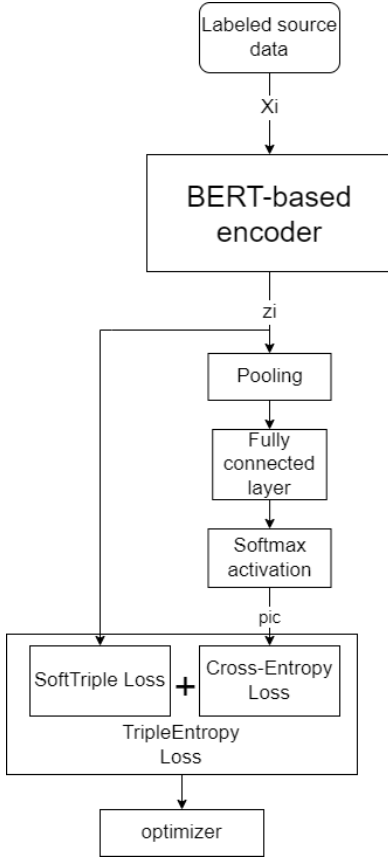


Fig. 1. BERT-based model fine-tuning architecture using TripleEntropy loss as the sum of SoftTriple Loss and Cross-Entropy Loss

C. Datasets

We conducted experiments to assess the usefulness of our TripleEntropy loss. To do so, we employed a variety of well-known datasets from SentEval [22] along with the IMDb [23]. These datasets cover both text classification and textual entailment as two important natural language tasks. Additionally, we have examined the performance of our method when the number of training examples is limited to 1,000 and 10,000 observations on sampled datasets. Table I shows the description of the datasets and their sampled versions.

IV. RESULTS

Our results are presented in the form of a comparison between the performance of the RoBERTa-base (RB) and the RoBERTa-large (RL) models as baselines, followed by the RoBERTa-base with TripleEntropy Loss (RB TripleEntropy) as well as RoBERTa-large with TripleEntropy Loss (RL TripleEntropy). All results shown below are expressed as a weighted F1 score. Moreover, we have created 4 experimental groups depending on the size of the dataset. In the first group, we present results regarding the small-sized datasets with a number of sentences of 1,000. In the second group, we explore results for the medium-sized datasets in which the number of sentences is about 4,000. In the third group, we present

results belonging to the large-sized datasets with a number of sentences of about 10,000. The extra-large-sized group consists of elements where the number of observations is larger than 50,000.

The RB baseline models were trained with the use of AdamW optimizer [30], beginning learning rate $1e-5$, L2 regularization, learning rate scheduler and linear warmup from 0 to $1e-5$ for the first 6% of steps and batch size of 64. The RB TripleEntropy models were trained on the same set of hyperparameters as the baseline models they refer to and additional parameters specific to TripleEntropy Loss, as it is described in Section III-B.

A. RB TripleEntropy for small datasets

Table II presents the results for the datasets containing 1,000 sentences. We observe that models trained using TripleEntropy have a higher performance than the baselines by about 0.71 percentage points. It is worth noting that the gain in performance is observed in each dataset, especially for the TREC-1k and MRPC-1k, where it amounts to 2.29 and 1.11 percentage points, respectively.

B. RB TripleEntropy for medium datasets

Table III shows the results based on the datasets containing about 4,000 sentences. Here, we can observe that models trained using TripleEntropy have higher performance than the baselines by about 0.86 percentage points. The highest gain in performance is observed in the case of TREC and MRPC datasets by 1.00 and 1.28 percentage points, respectively.

C. RB TripleEntropy for large datasets

Table IV shows the results based on the datasets containing about 10,000 sentences. The gain in the performance amounts to 0.20 percentage points.

D. RB TripleEntropy for extra-large datasets

Table V shows the results based on the datasets containing more than 50,000 sentences. The gain in the performance is not as high as in the case of the medium and small-sized datasets, and it is 0.04 percentage points on average, which is not significant.

E. RL TripleEntropy for small datasets

We have compared our results to the related work [21] where the authors claim the performance gains over baseline RoBERTa-large by applying loss function consisted of cross-entropy loss and Supervised Contrastive Learning loss. The work shows the improvement over baseline in the few-shot learning, defined as fine-tuning based on the training dataset consisting of 20, 100 and 1,000 observations. In order to compare our new loss function with the results from the related work, we conducted experiments where the baseline was RoBERTa-large (RL) with cross-entropy loss and compared it to the RoBERTa-large with TripleEntropy loss (RL TripleEntropy) on the dataset consisted of 1,000 observations. Our method yields a gain over baseline of 0.48 percentage points, which is higher than the performance improvement

TABLE I
SENTEVAL AND IMDB DATASETS, AND THEIR SAMPLED SUBSETS, USED IN OUR EVALUATION.

Dataset	# Sentences	# Classes	Sampled sub-sets	Task
SST2	67k	2	10k, 1k	Sentiment (movie reviews)[24]
IMDb	50k	2	10k, 1k	Sentiment (movie reviews) [23]
MR	11k	2	10k, 1k	Sentiment (movie reviews) [25]
MPQA	11k	2	10k, 1k	Opinion polarity [26]
SUBJ	10k	2	1k	Subjectivity status [27]
TREC	5k	6	4k, 1k	Question-type classification [25]
CR	4k	2	1k	Sentiment (product review) [28]
MRPC	4k	2	1k	Paraphrase detection [29]

TABLE II
WEIGHTED F1 SCORE OF ROBERTA-BASE (RB) VS ROBERTA-BASE WITH TRIPLEENTROPY LOSS (RB TRIPLEENTROPY) FOR SMALL DATASETS CONTAINING 1,000 OBSERVATIONS

Model	SST2-S	IMDb-S	SUBJ-S	MPQA-S	MRPC-S	TREC-S	CR-S	MR-s	avg
RB	88.63	81.00	94.61	87.75	78.01	79.80	91.57	85.89	85.91
RB TripleEntropy	89.09	81.45	94.70	87.93	79.12	82.09	92.16	86.39	86.62

TABLE III
WEIGHTED F1 SCORE OF ROBERTA-BASE (RB) VS ROBERTA-BASE WITH TRIPLEENTROPY LOSS (RB TRIPLEENTROPY) FOR MEDIUM DATASETS CONTAINING ABOUT 4,000 OBSERVATIONS

Model	MRPC-M	TREC-M	CR-M	avg
RB	83.11	96.19	93.28	90.86
RB TripleEntropy	84.39	97.19	93.58	91.72

TABLE IV
WEIGHTED F1 SCORE OF ROBERTA-BASE (RB) VS ROBERTA-BASE WITH TRIPLEENTROPY LOSS (RB TRIPLEENTROPY) FOR LARGE DATASETS CONTAINING ABOUT 10,000 OBSERVATIONS

Model	SST2-L	IMDb-L	SUBJ-L	MPQA-L	MR-L	avg
RB	92.63	85.12	96.83	91.08	89.09	90.95
RB TripleEntropy	92.79	85.23	97.15	91.30	89.29	91.15

TABLE V
WEIGHTED F1 SCORE OF ROBERTA-BASE (RB) VS ROBERTA-BASE WITH TRIPLEENTROPY LOSS (RB TRIPLEENTROPY) FOR EXTRA LARGE DATASETS CONTAINING MORE THAN 50,000 OBSERVATIONS

Model	SST2-XL	IMDb-XL	avg
RB	94.89	87.10	91.00
RB TripleEntropy	94.95	87.12	91.04

over baseline for a dataset of the same size from the related work, in which improvement over baseline is 0.27 percentage points. The results are presented in Table VI.

F. Discussion

Our method improves the performance most significantly for the small-sized dataset by 0.87 percentage points in the case of the RoBERTa-base baseline and 0.48 percentage points in the case of the RoBERTa-large baseline and the medium-sized dataset, where the increase amounts to 0.86 percentage points. For the large-sized dataset, the rise over baseline is 0.20%, while for the extra-large-sized dataset, the gain over baseline amounts to 0.04 percentage points. Our experiments show consistent performance improvement over baseline when using TripleEntropy loss, which is highest for the small and medium-sized datasets and decreases for the large and extra-large sized datasets. It is an improvement over previous related

work, where the performance improvement for the supervised classification tasks was achieved only for the few-shot learning settings [21].

We also conclude that the smaller the dataset is, the higher our new goal function’s performance gain over baseline. This observation is consistent with the conclusions of previous work [21]. The increase is negligible when the dataset is larger than about 10k observations. In addition, our work focuses on datasets of no less than 1k observations, so we do not know how it behaves in the case of few-shot learning, which in contrast, has been well documented in the case of work [21]. The performance comparison between baseline and our method throughout dataset size is depicted in Figure 2.

V. CONCLUSIONS

We proposed a supervised Distance Learning Metric objective that increases the performance of the RoBERTa-base

TABLE VI
WEIGHTED F1 SCORE OF ROBERTA-LARGE (RL) VS ROBERTA-LARGE WITH TRIPLEENTROPY LOSS (RL TRIPLEENTROPY) FOR SMALL DATASETS CONTAINING 1,000 OBSERVATIONS

Model	SST2-S	MPQA-S	MRPC-S	TREC-S	CR-S	MR-S	avg
RL	91.96	90.18	76.09	83.75	93.43	89.69	87.52
RL TripleEntropy	92.14	90.59	77.16	84.59	93.62	89.89	88.00

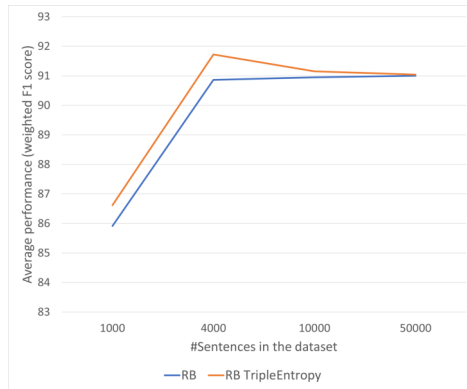


Fig. 2. Performance comparison between RB and RB TripleEntropy

models, which are strong baselines in the Natural Language Processing tasks. The performance is improved over multiple tasks from the single sentence classification and pair sentence classification to be higher by about 0.02-2.29 percentage points depending on the training dataset size. In addition, each result has been confirmed through tests with 5-fold cross-validation on 4 different seeds to increase its reliability.

In the future, we plan to investigate the effect of other DML methods on the performance of language models in a manner similar to the SoftTriple Loss method. We also want to extend the applicability of TripleEntropy by comparing the results with language models from different architectures, such as BERT, DistilBERT, or XLNet, to investigate its overall usefulness. Furthermore, given that our new loss function performs better the smaller the dataset, we plan to test how TripleEntropy behaves under few-shot learning settings.

ACKNOWLEDGMENTS

The research was funded by the Centre for Priority Research Area Artificial Intelligence and Robotics of Warsaw University of Technology within the Excellence Initiative: Research University (IDUB) programme (grant no 1820/27/Z01/POB2/2021).

REFERENCES

- [1] L. White, R. Togneri, W. Liu, and M. Bennamoun, "How well sentence embeddings capture meaning," in *Proceedings of the 20th Australasian document computing symposium*, 2015, pp. 1–8. [Online]. Available: <https://doi.org/10.1145/2838931.2838932>
- [2] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.
- [3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018. [Online]. Available: <https://doi.org/10.48550/arXiv.1810.04805>
- [4] Q. Qian, L. Shang, B. Sun, J. Hu, H. Li, and R. Jin, "Softtriple loss: Deep metric learning without triplet sampling," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 6450–6458. [Online]. Available: <https://doi.org/10.48550/arXiv.1909.05235>
- [5] C. Parsing, "Speech and language processing," 2009.
- [6] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013. [Online]. Available: <https://doi.org/10.48550/arXiv.1301.3781>
- [7] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019. [Online]. Available: <https://doi.org/10.48550/arXiv.1907.11692>
- [8] S. Dadas, M. Perelkiewicz, and R. Pościwata, "Pre-training polish transformer-based language models at scale," in *International Conference on Artificial Intelligence and Soft Computing*. Springer, 2020, pp. 301–314. [Online]. Available: https://doi.org/10.1007/978-3-030-61534-5_27
- [9] K. Q. Weinberger and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification." *Journal of machine learning research*, vol. 10, no. 2, 2009.
- [10] E. Xing, M. Jordan, S. J. Russell, and A. Ng, "Distance metric learning with application to clustering with side-information," *Advances in neural information processing systems*, vol. 15, pp. 521–528, 2002.
- [11] S. Wu, X. Feng, and F. Zhou, "Metric learning by similarity network for deep semi-supervised learning," in *Developments of Artificial Intelligence Technologies in Computation and Robotics: Proceedings of the 14th International FLINS Conference (FLINS 2020)*. World Scientific, 2020, pp. 995–1002. [Online]. Available: <https://doi.org/10.48550/arXiv.2004.14227>
- [12] Y. Movshovitz-Attias, A. Toshev, T. K. Leung, S. Ioffe, and S. Singh, "No fuss distance metric learning using proxies," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 360–368. [Online]. Available: <https://doi.org/10.48550/arXiv.1703.07464>
- [13] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *European conference on computer vision*. Springer, 2016, pp. 499–515. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-46478-7_31
- [14] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, vol. 2. IEEE, 2006, pp. 1735–1742. [Online]. Available: <https://doi.org/10.1109/CVPR.2006.100>
- [15] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 815–823. [Online]. Available: <https://doi.org/10.1109/CVPR.2015.7298682>
- [16] "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PMLR, 2020, pp. 1597–1607. [Online]. Available: <https://doi.org/10.48550/arXiv.2002.05709>
- [17] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," *arXiv preprint arXiv:1703.07737*, 2017.
- [18] B. Skuczyńska, S. Shaar, J. Spenader, and P. Nakov, "Beasku at checkthat! 2021: fine-tuning sentence bert with triplet loss and limited data," *Faggioli et al.[33]*, 2021.
- [19] I. Malkiel, D. Ginzburg, O. Barkan, A. Caciularu, Y. Weill, and N. Koenigstein, "Metricbert: Text representation learning via self-supervised triplet training," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 1–5. [Online]. Available: <https://doi.org/10.1109/ICASSP43922.2022.9746018>

- [20] M. Lennox, N. Robertson, and B. Devereux, "Deep learning proteins using a triplet-bert network," in *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE, 2021, pp. 4341–4347. [Online]. Available: <https://doi.org/10.1109/embc46164.2021.9630387>
- [21] B. Gunel, J. Du, A. Conneau, and V. Stoyanov, "Supervised contrastive learning for pre-trained language model fine-tuning," *arXiv preprint arXiv:2011.01403*, 2020. [Online]. Available: <https://doi.org/10.48550/arXiv.2011.01403>
- [22] A. Conneau and D. Kiela, "Senteval: An evaluation toolkit for universal sentence representations," *arXiv preprint arXiv:1803.05449*, 2018. [Online]. Available: <https://doi.org/10.48550/arXiv.1803.05449>
- [23] A. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," in *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, 2011, pp. 142–150.
- [24] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proceedings of the 2013 conference on empirical methods in natural language processing*, 2013, pp. 1631–1642.
- [25] B. Pang and L. Lee, "Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales," *arXiv preprint cs/0506075*, 2005. [Online]. Available: <http://dx.doi.org/10.3115/1219840.1219855>
- [26] J. Wiebe, T. Wilson, and C. Cardie, "Annotating expressions of opinions and emotions in language," *Language resources and evaluation*, vol. 39, no. 2, pp. 165–210, 2005. [Online]. Available: <https://doi.org/10.1007/s10579-005-7880-9>
- [27] B. Pang and L. Lee, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," *arXiv preprint cs/0409058*, 2004. [Online]. Available: <https://doi.org/10.3115/1218955.1218990>
- [28] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2004, pp. 168–177. [Online]. Available: <https://doi.org/10.1145/1014052.1014073>
- [29] W. Dolan, C. Quirk, C. Brockett, and B. Dolan, "Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources," 2004.
- [30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014. [Online]. Available: <https://doi.org/10.48550/arXiv.1412.6980>

About Classifiers Quality Assessment: Balanced Accuracy Curve (BAC) as an Alternative for ROC and PR Curve

Aleksandra Weiss, Marcin Młyński
 Scientific Circle of Robotics UWM in Olsztyn
 ul. Słoneczna 54, 10-710 Olsztyn, Poland
 Email: {aleksandra.weiss28, marcinmlynski920}@gmail.com

Piotr Artiemjew
 University of Warmia and Mazury,
 in Olsztyn
 ul. Słoneczna 54, 10-710 Olsztyn, Poland
 Email: artem@matman.uwm.edu.pl

Abstract—In this work, we propose a new parameter to study the effectiveness of classifiers - the AUC (area under curve) of the balanced accuracy curve (BAC) on data with different balance degrees - we compare its effectiveness with the popular AUC parameters for the ROC and PR curve. We use a global kNN classifier with typical metrics to verify the utility of the new parameter. BAC, ROC and PR curves generate similar results, the advantage of BAC is its simplicity of implementation and ease of interpretation of results.

I. INTRODUCTION

CLASSIFICATION accuracy is the most natural parameter for assessing classification quality. Total accuracy fails when test data are unbalanced in terms of class sizes. With help comes a balanced version of accuracy, which is simply the efficiency of the average across all test classes. Thanks to this parameter, even small test classes are equally taken into account in the classification. In this work, we present a preliminary verification of whether the AUC of a balanced accuracy curve applied on training data balanced in different levels can create a valuable competitive parameter for PR and ROC curve. Let us turn to a brief review of the literature on the topic under discussion. First of all, the parameters that we discuss in this paper concern the evaluation of binary classifiers. Let us introduce the basic notation. The following symbols shall be used: $T = True$, $P = Positive$, $F = False$, $N = Negative$. TP is the number of test objects from the positive class that were correctly classified. FP is the number of test objects from the negative class that were classified into the positive class. FN is the number of objects in the positive class that have been classified in the negative class. A receiver operating characteristic curve (ROC) is a chart that shows the predictive capability of a binary classifier as its threshold of discrimination evolves. The method was initially designed for military radar receiver operators in 1941, resulting in its name [1], [2]. The ROC curve shows the ratio of TP to FP values through the prism of thresholds determining membership to a positive class. The best value of the ROC curve is closest to the upper left corner of the plot. A Precision Recall curve (PR) [4], [3] is basically a chart with the Precision values on the y axis and the Recall on the x axis. $precision = \frac{TP}{TP+FP}$

Precision is understood as the accuracy of the classification of a positive class - the percentage of correctly classified objects in that class out of those classified. $recall = \frac{TP}{TP+FN}$ The Recall parameter tells us about the relevance to the positive class - it specifies the percentage of correctly classified objects in the positive class in relation to all classified objects to the positive class. The best value of the PR curve is closest to the top right corner of the plot. An extensive introduction of the relationship of ROC and PR curves can be seen in papers [5] and [6]. In paper [7], the author leads a discussion on the imbalance of decision classes vs PR curve. The literature review related to classification quality assessment is enormous, we will not even attempt to review it in this paper, for further reading the reader is directed to e.g. paper [12].

In the following sections we have the following content. In Section II we present the research methodology. In section III we present the results of the experiments. In section IV we summarise the work and indicate further research plans. Let us move on to discuss the methodology used in this thesis.

II. METHODOLOGY

In this section we discuss how we implemented the ROC, PR, BAC curves and introduce information about the classifier used.

A. Basic classifier

In testing the effectiveness of AUC values for BAC, ROC and PR curve, we used the kNN classifier. Which does not mean that there is any restriction on the use of other classifiers. The one chosen is an initial reference point. The procedure used is as follows.

Step 1. We input a training decision system (U^{trn}, A, d) and a test decision system (U_{tst}, A, d) , where A is a set of conditional attributes, d a decision attribute.

Step 2. The classification of test objects using training objects is carried out as follows.

Across all conditional attributes $a \in A$, training objects $v \in U^{trn}$ and test objects $u \in U_{tst}$, we calculate the importance weights $w(u, v)$ using selected metric.

In the version used, the obvious limitation k is the size of the training system.

The test object u is classified using the weights computed for all training objects v . The weights are ordered in ascending order as,

$$w_1(u, v_1) \leq w_2(u, v_2) \leq \dots \leq w_{|U_{trn}|}(u, v|_{U_{trn}})$$

Using the calculated and sorted weights, the training decision classes vote with the following parameter, where c is over the decision classes in the training set,

$$Class_{weight_c}(u) = \sum_{i=1}^k w_i^c(u, v_i^c).$$

Eventually, the test object u is categorized into the class c with the smallest value $Class_{weight_c}(u)$.

Assume that the positive class is denoted by 1 and the negative class by 0. Once all test objects u have been classified, the quality parameter accuracy, acc is calculated, according to the equation

$$acc_{balanced} = \frac{acc_{class_1} + acc_{class_0}}{2}$$

$$acc_{class_c} = \frac{|correctly\ classified\ objects\ in\ class_c|}{|classified\ objects\ in\ class_c|}$$

B. BAC curve

We propose the AUC of the balanced accuracy curve as a new factor for cross-sectional assessment of classification quality. The curve is constructed from balanced accuracy values using training systems with varying levels of class balance. We divide the analyzed data into test and training datasets and determine which of the two possible classes is a positive class and which is a negative class. After classifying each decision class, we create a test set containing 30% of the total number of objects. From the remaining 70% (excluding the data contained in the test set), we create a training set. The splitting and classification procedure is carried out a hundred times - we include the average accuracy value as the final result. In the case of BAC, we use a 0.5 threshold for classification. The class balance levels (of training decision system) we use are: (10 : 90, 20 : 80, 30 : 70...90 : 10). The use of BAC is possible when the decision classes are adequately represented in terms of size, since we require the creation of subsets with different levels of balance used data should have access to the same number of objects in the classes. In other words, in the case of data that is highly unbalanced, where one of the important classes is small the use of BAC can be difficult.

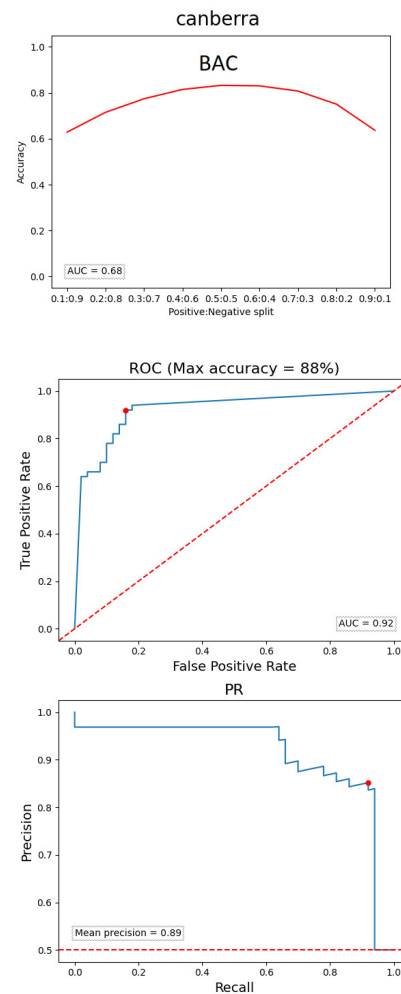


Fig. 1. An exemplary detailed result for Australian Credit decision system. From the top, BAC, ROC, and PR curves. The metric used is Canberra defined as follows: $d(x, y) = \sum_{i=1}^n \frac{|x_i - y_i|}{|x_i| + |y_i|}$

C. ROC and PR curve

We chose the AUC of the ROC and PR curves as reference parameters. We divide the analyzed data into test and training datasets and determine which of the two possible classes is the positive class and which is the negative class. After determining each decision class, we select 50 random objects from each class and combine the objects into one test set containing 100 objects. Then using the remaining objects from the entire dataset, we create a training set. We use the classification probabilities obtained after applying the kNN classifier to obtain the optimal threshold visualized by ROC curve and PR curve. We compare the probabilities with thresholds in the range from 0 to 1 with a step=0.001. If the probability is greater than or equal to the threshold, then we classify the object into the positive class, otherwise into the negative class. For each threshold, we calculate the coefficients of true positives (sensitivity), false positives and the accuracy

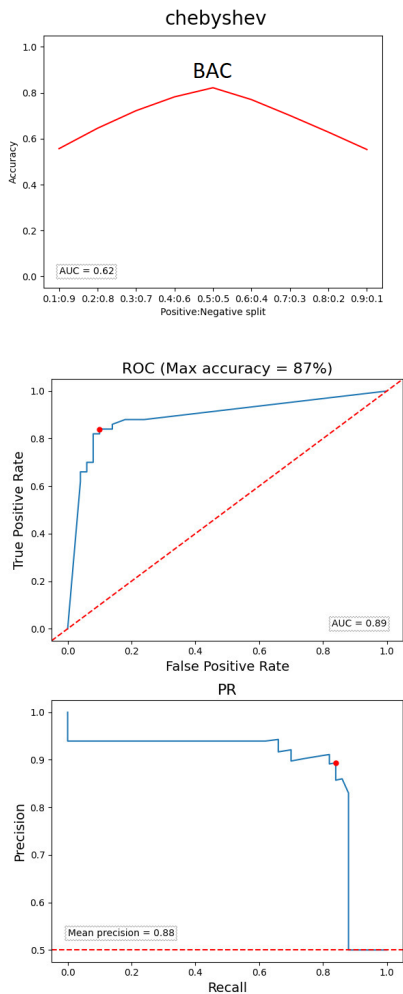


Fig. 2. An exemplary detailed result for Australian Credit decision system. From the top, BAC, ROC, and PR curves. The metric used is Chebyshev distance defined as follows: $d(x, y) = \max_i |x_i - y_i|$

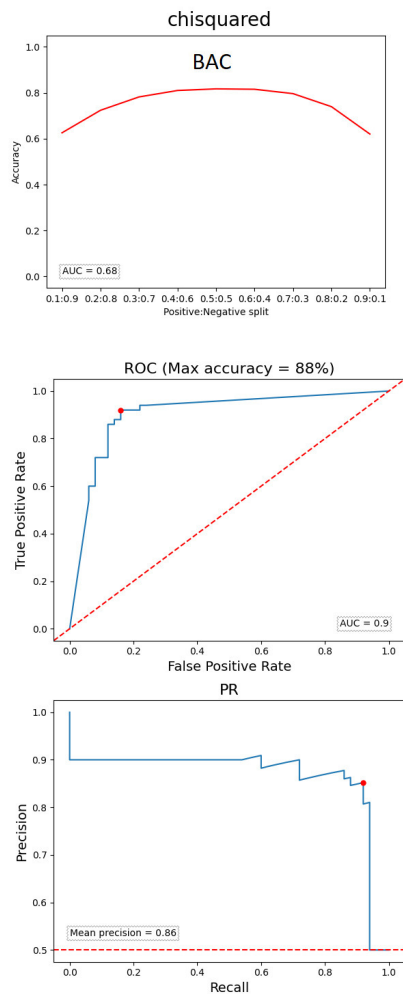


Fig. 3. An exemplary detailed result for Australian Credit decision system. From the top, BAC, ROC, and PR curves. The metric used is Chi-squared distance defined as follows: $d(x, y) = \sum_{i=1}^n \left(\frac{|x_i - y_i|}{|x_i| + |y_i|} \right)^2$

(precision) of classifying objects. The example detailed results for the ROC and PR curves, due to the difficulty of showing the mean result are from single classifications. The results summarising the performance of the curves are the average of a hundred times of the experiment.

III. EXPERIMENTAL SESSION

In the experimental part we use the kNN method (See description in section II-A) with metrics: manhattan, euclidean, manhattan-maxmin, cosine, minkowski, chebyshev, canberra, chi-square, jaccard, epsilonHamming, sørensen. Definitions of each metric are provided in the headings of Figures 1 to 11. For the selected classifier, custom implementations for creating BAC, ROC and PR curves were written. We verify parameter performance on three decision systems selected from the UCI repository [15] - Australian Credit, Pima Indians Diabetes and Heart Disease datasets. Detailed results showing all three

curves are presented for the Australian Credit decision system only - see figures from 1 to 11. A summary of metrics ranking generation based on AUC of BAC, ROC and PR curves for Australian Credit, Pima Indians Diabetes and Heart Disease systems can be seen in figures 12, 13 and 14.

A. Result summary

In Figures 12, 13 and 14 we have the rankings of the kNN method metrics based on the AUC of the BAC, ROC and PR curves - for Australian Credit, Heart Disease and Pima Indians Diabetes datasets respectively.

Results for Australian Credit decision system : AUC range of three positions for BAC and ROC curve agree for the metrics: canberra, euclidean, manhattan, minkowski, sørensen and epsilonHamming. AUC range of three positions for BAC and PR curve agree for the metrics: canberra, euclidean, minkowski, sørensen, epsilonHamming.

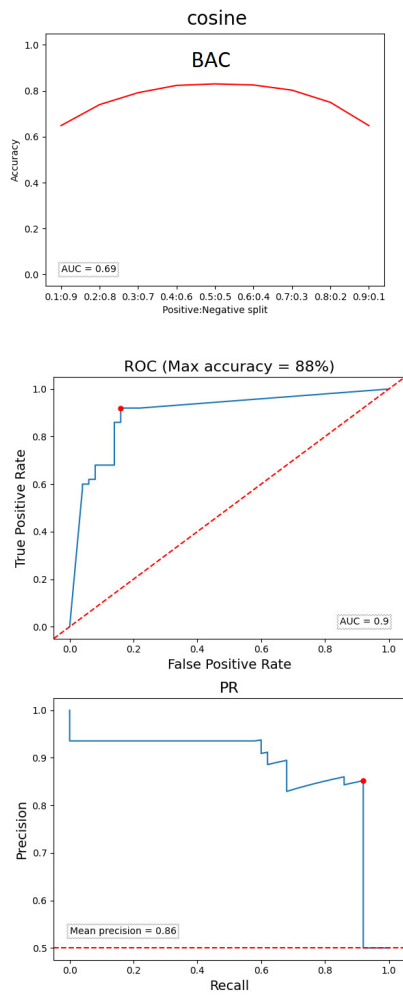


Fig. 4. An exemplary detailed result for Australian Credit decision system. From the top, BAC, ROC, and PR curves. The metric used is Cosine distance

defined as follows:
$$d(x, y) = \frac{\sum_{i=1}^n x_i y_i}{\sqrt{\sum_{i=1}^n x_i^2} \sqrt{\sum_{i=1}^n y_i^2}}$$

Results for Heart Disease decision system: AUC range of three positions for BAC and ROC curve agree for the metrics: canberra, cosine, chisquared, manhattan, maxmin, euclidean, minkowski, chebyshev, epsilonHamming and sorenson. AUC range of three positions for BAC and PR curve agree for the metrics: jaccard, canberra, manhattan, maxmin, chisquared, minkowski, epsilonHamming, sorenson.

Results for Pima Indians Diabetes decision system: AUC range of three positions for BAC and ROC curve agree for the metrics: jaccard, cosine, chisquared, maxmin, sorenson and epsilonHamming. AUC range of three positions for BAC and PR curve agree for the metrics: jaccard, manhattan, chisquared, maxmin, minkowski, sorenson and epsilonHamming. The overall shape of the ranking indicates that the BAC, ROC and PR curves are comparable tools for assessing classification quality.

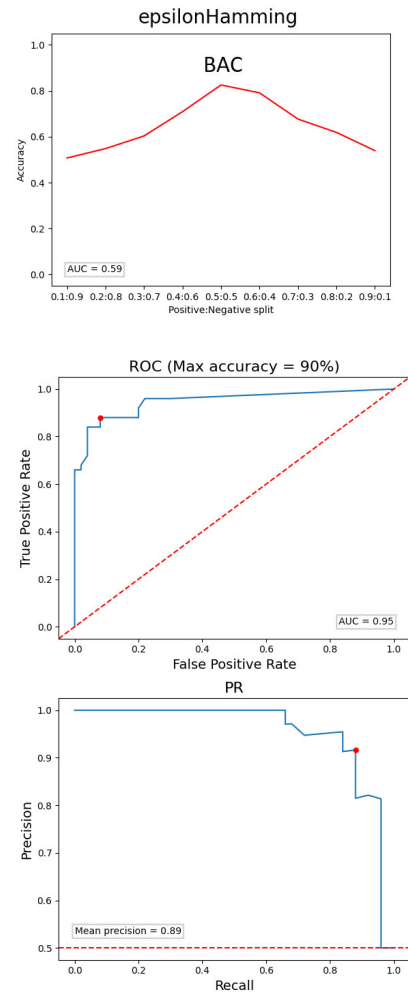


Fig. 5. An exemplary detailed result for Australian Credit decision system. From the top, BAC, ROC, and PR curves. The metric used is epsilonHamming distance

defined as follows:
$$d(x, y) = |\{a \in A : \text{dist}(a(x_i), a(y_i)) \geq \varepsilon\}|, \varepsilon = 0.01$$

When analyzing the results for the three selected decision systems (Australian Credit, Heart Disease, and Pima Indians Diabetes), we see that the area under the balanced accuracy curve for the entire training system balance spectrum can be a competitive factor for determining the quality of classifiers to ROC and PR curve. The ranking results are similar to each other. Some metrics are mixed among themselves because the data are randomly selected, but there are clear common features to these rankings. The shape of the curve showing the ranking of metrics is similar. The main advantage of using the field under the accuracy curve is the simplicity of implementation and the ease of understanding the resulting factor. Which is simply the cross-sectional stability of the classifier for training knowledge balanced in different levels.

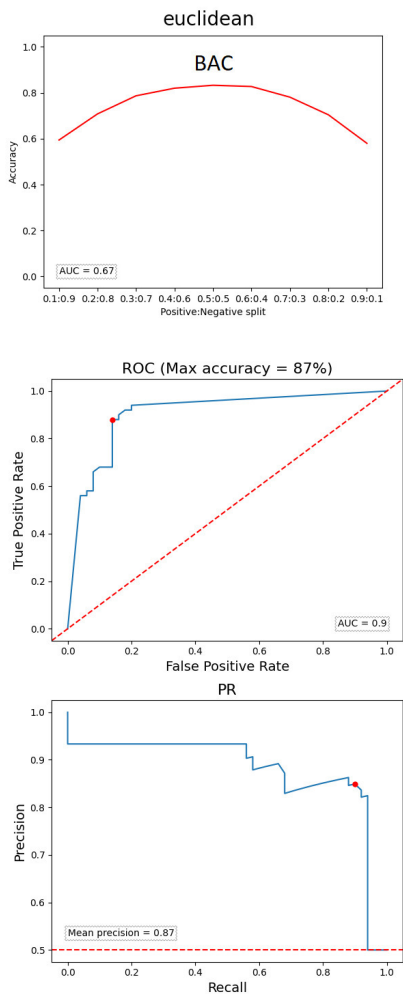


Fig. 6. An exemplary detailed result for Australian Credit decision system. From the top, BAC, ROC, and PR curves. The metric used is euclidean distance defined as follows: $d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$

IV. CONCLUSION

This paper examines the applicability of the field under the accuracy curve-defined as the set of classification accuracies using training systems with different levels of decision class balancing-to the cross-sectional evaluation of the kNN classifier. We used the global method (k is selected from the entire system at once) with different metrics as reference kNN variants. By seeing the results, we can summarise that the specified factor is competitive with the ROC and PR curve, with its implementation and interpretation being much simpler and more understandable. Preliminary results show a similar gradation of methods, the difference in the performance of the metrics is due to the fact that the data were randomly selected, but the overall ranking looks similar. We consider our results as an initial step to open a discussion on the potential applicability of BAC to assess the stability of classifiers. We plan to extend our research to the entire spectrum of

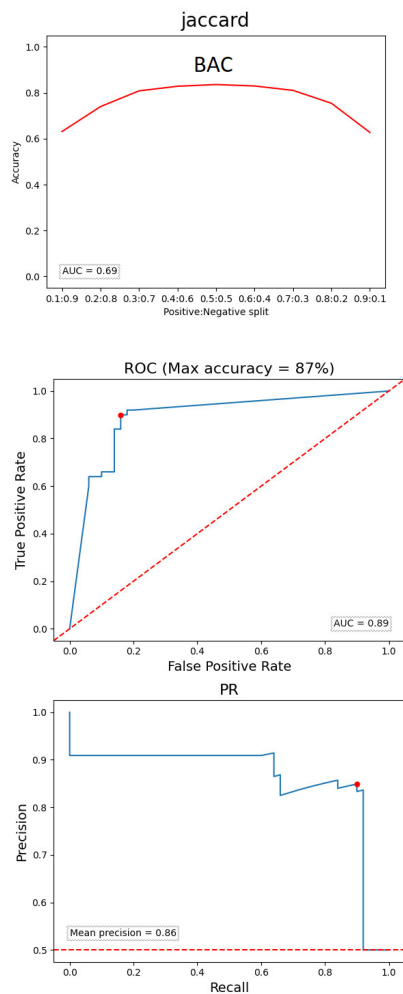


Fig. 7. An exemplary detailed result for Australian Credit decision system. From the top, BAC, ROC, and PR curves. The metric used is jaccard distance defined as follows: $d(x, y) = 1 - \left| \frac{\sum_{i=1}^n (x_i * y_i)}{\sum_{i=1}^n x_i^2 + \sum_{i=1}^n y_i^2 - \sum_{i=1}^n (x_i * y_i)} \right|$

classification methods in the future.

ACKNOWLEDGMENT

This work has been supported by the grant from Ministry of Science and Higher Education of the Republic of Poland under the project number 23.610.007-000

REFERENCES

- [1] Woodward, P. M. (1953). Probability and information theory with applications to radar. London: Pergamon Press.
- [2] Peterson, W., Birdsall, T., Fox, W. (1954). The theory of signal detectability, Transactions of the IRE Professional Group on Information Theory, 4, 4, pp. 171 - 212.
- [3] Manning, C., & Schütze, H. (1999). Foundations of statistical natural language processing. MIT Press
- [4] Raghavan, V., Bollmann, P., & Jung, G. S. (1989). A critical investigation of recall and precision as measures of retrieval system performance. ACM Trans. Inf. Syst., 7, 205-229.

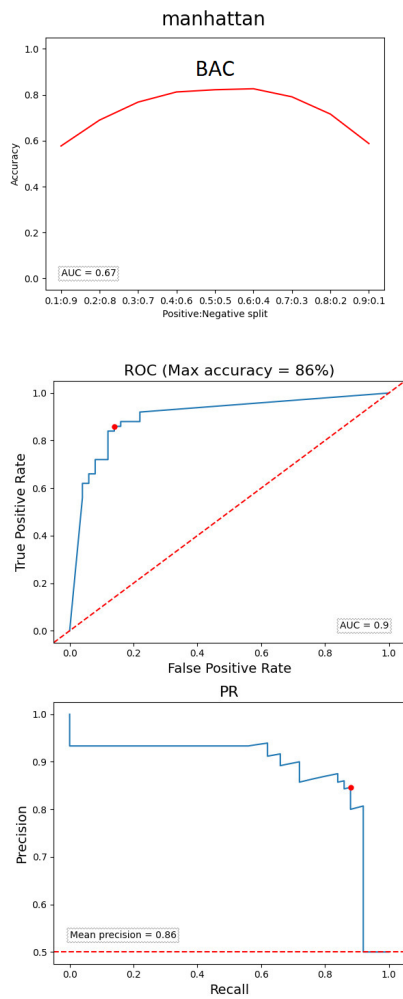


Fig. 8. An exemplary detailed result for Australian Credit decision system. From the top, BAC, ROC, and PR curves. The metric used is manhattan distance defined as follows: $d(x, y) = \sum_{i=1}^n |x_i - y_i|$

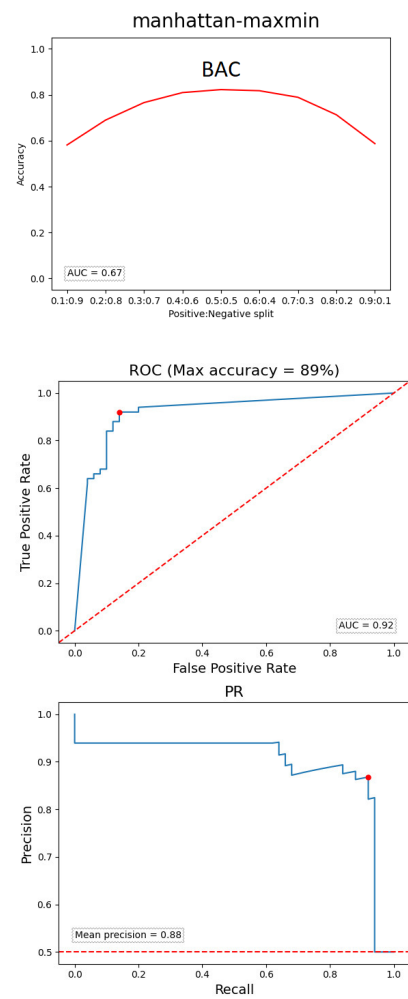


Fig. 9. An exemplary detailed result for Australian Credit decision system. From the top, BAC, ROC, and PR curves. The metric used is manhattan-maxmin distance defined as follows: $d(x, y) = \sum_{i=1}^n \sum_{j=1}^n \frac{|x_{ij} - y_{ij}|}{\max_j y_{ij}}$

- [5] Davis, J., Goadrich, M.: 2006. The relationship between Precision-Recall and ROC curves. In Proceedings of the 23rd international conference on Machine learning (ICML '06). Association for Computing Machinery, New York, NY, USA, 233–240. <https://doi.org/10.1145/1143844.1143874>
- [6] Saito T., and Rehmsmeier M. 2015. "The Precision-Recall Plot Is More Informative Than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets." PLoS ONE. 10(3): e0118432
- [7] Williams, C.K.I. 2021. "The Effect of Class Imbalance on Precision-Recall Curves." Neural Computation 33(4): 853–857.
- [8] Morzy, Tadeusz. Eksploracja danych. Red. . Warszawa: Wydawnictwo Naukowe PWN, 2013, 566 s. ISBN 978-83-01-17175-9
- [9] Hastie T., Friedman J., Tibshirani R. (2001) The Elements of Statistical Learning. Springer Series in Statistics. Springer, New York, NY.
- [10] Qimin Cao, Lei La, Hongxia Liu, and Si Han. Mixed Weighted KNN for Imbalanced Datasets [J]. Int J Performability Eng, 2018, 14(7): 1391-1400.
- [11] L., Polkowski, P., Artiemjew, "Granular Computing in Decision Approximation - An Application of Rough Mereology," in: Intelligent Systems Reference Library 77, Springer, ISBN 978-3-319-12879-5, 2015, pp. 1-422.
- [12] Japkowicz, N., & Shah, M. (2011). Evaluating Learning Algorithms: A Classification Perspective. Cambridge: Cambridge University Press. doi:10.1017/CBO9780511921803
- [13] Metrics definition: manhattan, euclidean, canberra, cosine <https://www.itl.nist.gov/div898/software/dataplot/homepage.htm>
- [14] epsilonHamming Metric definition: In: Polkowski, L., Artiemjew, P.: Granular Computing in Decision Approximation - An Application of Rough Mereology, In: Intelligent Systems Reference Library 77, Springer, ISBN 978-3-319-12879-5, pp. 1–422 (2015).
- [15] UCI Machine Learning Repository, <https://archive.ics.uci.edu/ml/index.php>. Last accessed 12 Apr 2022

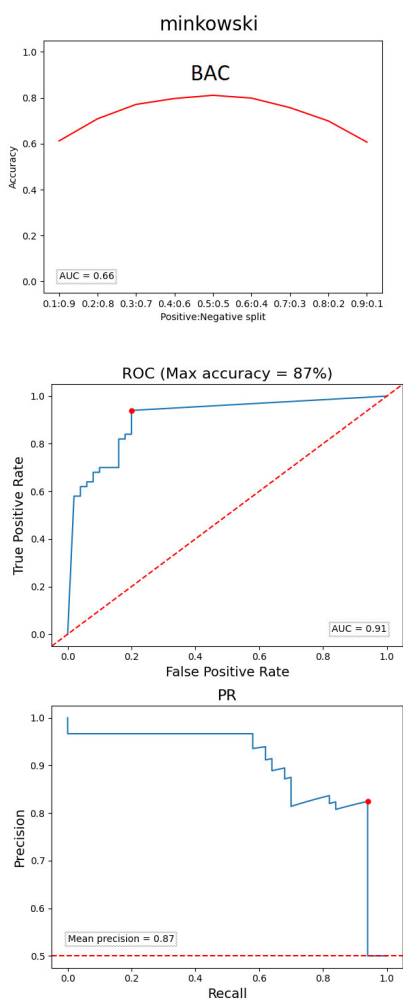


Fig. 10. An exemplary detailed result for Australian Credit decision system. From the top, BAC, ROC, and PR curves. The metric used is Minkowski distance defined as follows: $d(x, y) = \left(\sum_{i=1}^n |x_i - y_i|^p \right)^{\frac{1}{p}}$, $p = 3$

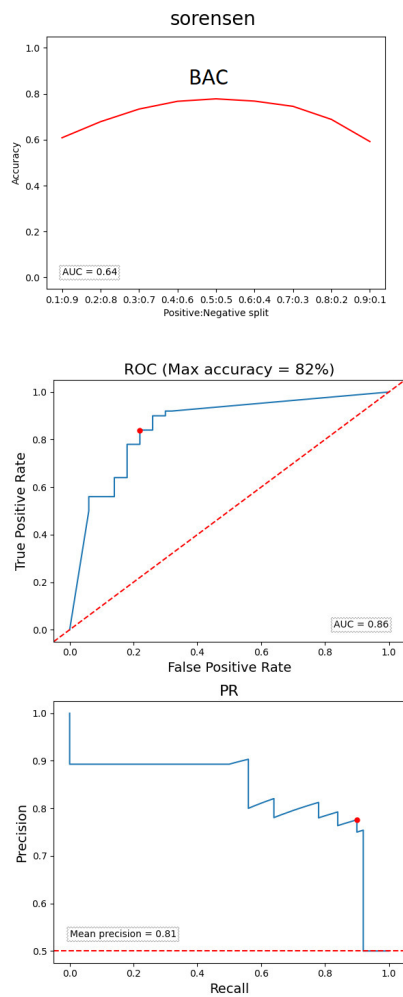


Fig. 11. An exemplary detailed result for Australian Credit decision system. From the top, BAC, ROC, and PR curves. The metric used is sorensen distance defined as follows: $d(x, y) = 1 - \left| \frac{2 * \sum_{i=1}^n (x_i * y_i)}{\sum_{i=1}^n x_i^2 + \sum_{i=1}^n y_i^2} \right|$

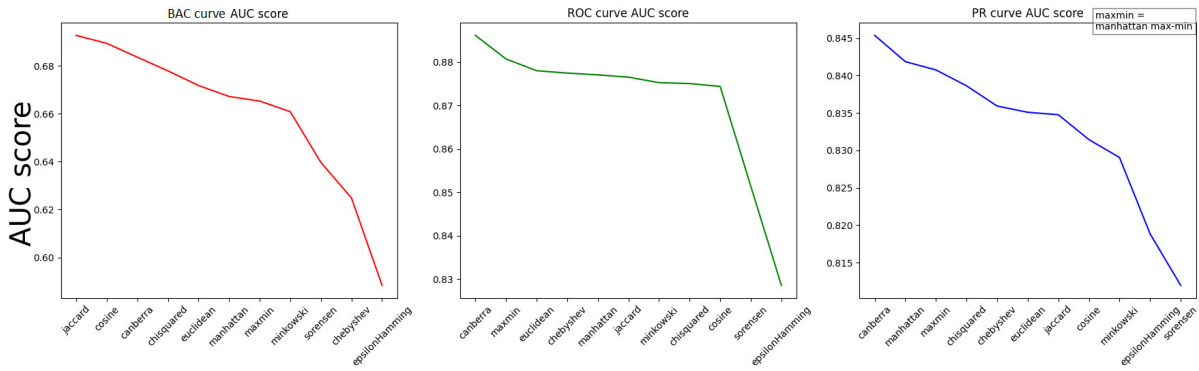


Fig. 12. Ranking of metrics for the Australia Credit data set. Comparison of rankings for AUC of BAC, ROC and PR curves.

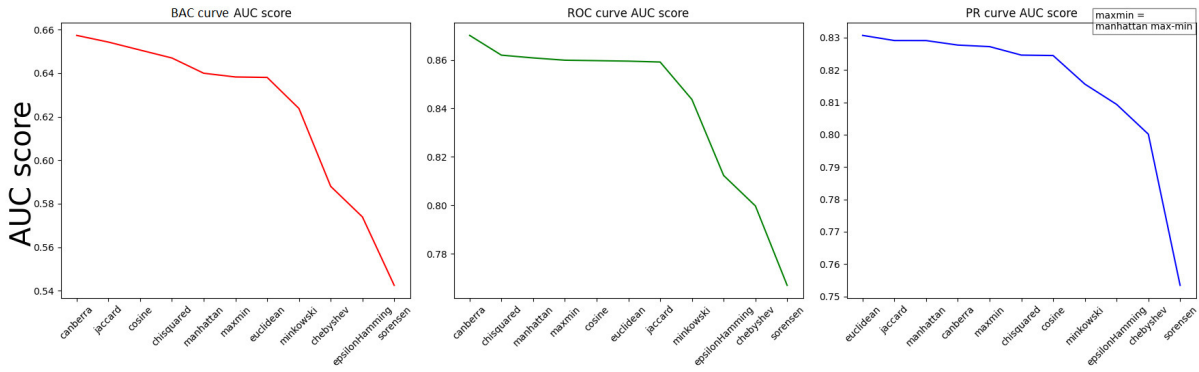


Fig. 13. Ranking of metrics for the Heart Disease data set. Comparison of rankings for AUC of BAC, ROC and PR curves.

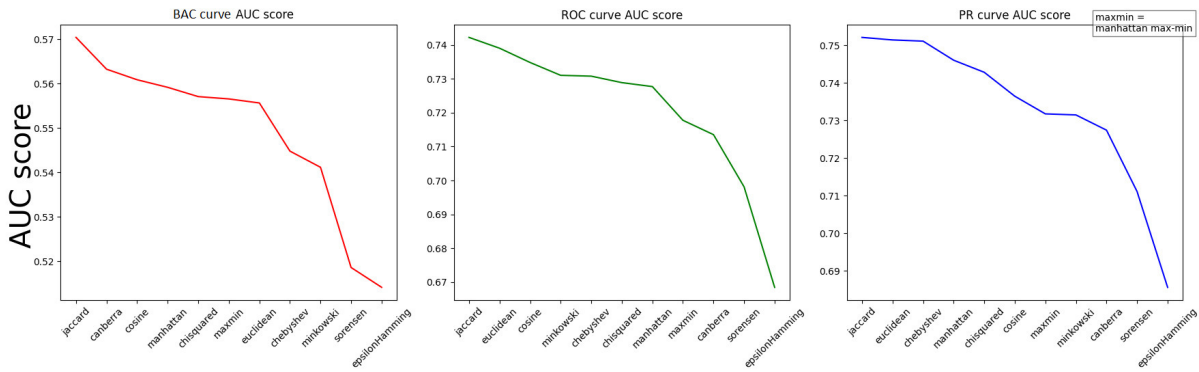


Fig. 14. Ranking of metrics for the Pima Indians Diabetes data set. Comparison of rankings for AUC of BAC, ROC and PR curves.

Extending Word2Vec with Domain-Specific Labels

Miloš Švaňa

VSB - Technical University of Ostrava

Department of Systems Engineering

17. listopadu 2172/15, 708 00 Ostrava-Poruba, Czechia

Email: milos.svana@vsb.cz

Abstract—Choosing a proper representation of textual data is an important part of natural language processing. One option is using Word2Vec embeddings, i.e., dense vectors whose properties can to a degree capture the "meaning" of each word. One of the main disadvantages of Word2Vec is its inability to distinguish between antonyms. Motivated by this deficiency, this paper presents a Word2Vec extension for incorporating domain-specific labels. The goal is to improve the ability to differentiate between embeddings of words associated with different document labels or classes. This improvement is demonstrated on word embeddings derived from tweets related to a publicly traded company. Each tweet is given a label depending on whether its publication coincides with a stock price increase or decrease. The extended Word2Vec model then takes this label into account. The user can also set the weight of this label in the embedding creation process. Experiment results show that increasing this weight leads to a gradual decrease in cosine similarity between embeddings of words associated with different labels. This decrease in similarity can be interpreted as an improvement of the ability to distinguish between these words.

I. INTRODUCTION

TRANSFORMATION of text into a numerical representation is an important part of any natural language processing (NLP) problem. Considering the level of words, one can choose between alternatives ranging from simple one-hot encoding to complex language models such as ELMo [9], BERT [3], or GPT-3 [2]. Word2Vec embeddings lie in-between these two extremes in terms of level of complexity.

Word embeddings are dense vectors usually of several hundred dimensions with the ability to at least partly capture the meaning of each word. This meaning is based on each word's context, i.e., words that co-occur with a given target word. It is captured by the relative position of different words in the embedding vector space. Words with similar meaning should be represented by vectors close to each other, while dissimilar words should be more distant. Moreover, basic operations such as addition or subtraction enable the derivation of representations for new words. A common example is the subtraction of the embedding for the word *man* from of the word *king*, followed by the addition of the embedding of *woman*. the result should be close to the embedding of the word *queen*.

Originally proposed by [6], Word2Vec is an algorithm for creating word embeddings. It is based on a simple idea: words with similar meaning occur in similar contexts. This context is usually defined by surrounding words. One issue with this line of thinking is that although Word2Vec works well for detecting

synonyms, hyponyms or hypernyms [5, 14], it can't easily distinguish between two antonyms [11, 1]. Hence, words such as *good* and *bad* are often represented by relatively similar embeddings.

Several authors, including [8] or [4], addressed this issue by extending the basic Word2Vec algorithm to consider not only the context of words, but also thesauri information. In cited papers, several well known public datasets such as WordNet or Roget are used. [10] claim that the information about antonyms can be extracted from the geometry of the embedding space with a method called *contrasting maps*.

This paper describes a simple modification of the Word2Vec algorithm for considering domain specific document labels data during embedding creation. In contrast with the aforementioned work, presented approach is more flexible and can be adopted to many domains of interest.

In the paper, the modification is applied in the domain of finance. The problem can be stated as follows: We have set of tweets related to a specific publicly traded company. It is assumed that certain words occur more frequently when the stock price of a company drops, while others are used more when the stock price rises. In many financial analysis tasks it would be useful to represent words with occurrences associated with these opposite situations with vectors that are distant from each other. Can we improve on the basic Word2Vec algorithm by considering the information about stock price increase or decrease in the time of tweet publication?

Incorporating this information directly into word embeddings could be helpful in risk or return prediction. Some researchers, e.g. [12] or [13] are already using the basic Word2Vec model for similar tasks and this work could improve the prediction power of their models.

The remainder of this paper is organized as follows: Section II provides a basic overview of the Word2Vec algorithm, section III describes a modification for considering domain-specific labels during embedding creation, and section IV introduces the experiments performed to evaluate this modification. Results of these experiments are then presented and discussed in section V. Finally, the presented work is summarized in section VI. This section also discusses future research opportunities.

II. WORD2VEC OVERVIEW

The Word2Vec algorithm is based on a simple feed-forward neural network with a single hidden layer. The number of

neurons in this hidden layer determines the dimensionality of the embedding space. Word embeddings are found by training this neural network in one of two ways – **continuous bag of words** (CBoW) and **skip-gram**.

The extension presented in this paper is based on the skip-gram model. When using this approach, the neural network uses the one-hot encoded target word as its input and tries to predict its multi-hot encoded context. The objective of the skip-gram model can be further formulated as maximizing the log probability [7]:

$$\frac{1}{T} \sum_{t=1}^T \sum_{-c \leq j \leq c, j \neq 0} \log p(w_{t+j}|w_t) \quad (1)$$

where T is the total number of target words, and c is the context window size, i.e., the number of words before or after the target word in the text to consider as skip-gram output.

The training sets for both CBoW and skip-gram methods are practically identical and can be created easily from a corpora of text documents. After the neural network is trained, the embeddings of different words are simply the weights of connections between the input element representing a given word in one-hot encoding and all neurons of the hidden layer.

In their subsequent paper, [7] extended the Word2Vec algorithm to improve both its performance in terms of training time and its accuracy. Two modifications were proposed:

- **Frequent word subsampling:** Words that occur frequently in the text are omitted from the document with probability:

$$P(w_i) = 1 - \sqrt{\frac{t}{f(w_i)}} \quad (2)$$

where $f(w_i)$ is the frequency of the word w_i and t is a threshold usually around 10^{-5} . Most frequent words include articles such as *the*, *a* and *an* that do not carry that much information about the contextual meaning of a specific word.

- **Negative sampling:** Leads to a modified objective function:

$$\log \sigma(v'_{w_o}{}^T v_{w_t}) + \sum_{i=1}^k \mathbb{E}_{w_i \sim P_n(w)} [\log \sigma(-v'_{w_i}{}^T v_{w_t})] \quad (3)$$

which in practical terms causes only k words that are missing in a given context to update their embeddings during training.

Experimental results show that these extensions lead to both more efficient training, and to better quality of final embeddings. Proposed extension implements word subsampling, but not negative sampling.

III. PROPOSED EXTENSION

As discussed in introduction, one of Word2Vec's disadvantages is antonym representation. Antonyms such as *good*

and *bad* are often surrounded by similar words, hence their Word2Vec embeddings are relatively similar.

I propose an extension of the Word2Vec model that allows for consideration of domain specific document labels to better distinguish between words related to different classes. In contrast to previous work presented in section I, it does not rely on general purpose thesauri.

The extension is based on the idea of modifying the output to be predicted by the neural network. In addition to context (surrounding words) for a certain input word, the neural network also has to predict the document class.

This extension further lets the user set the weight of this class label prediction relative to word context prediction. Modified objective function can then be formulated as:

$$\frac{1}{T} \sum_{t=1}^T \sum_{-c \leq j \leq c, j \neq 0} (\log p(w_{t+j}|w_t) + u \log p(d_t|w_t)) \quad (4)$$

where u is the label prediction weight and d_t is the document label associated with word w_t . This weight is implemented as replication of the output neurons for label prediction u times.

IV. EXPERIMENT DESIGN

To verify that the extension helps the Word2Vec algorithm better consider domain specific labels, an experiment with the goal of creating embeddings from tweets related to a publicly traded company was performed. Each tweet was labeled with a label "1" if the stock price increased one day after tweet publication. If the stock price decreases, the tweet was labeled with "0". Difference of 2 subsequent daily close prices was used to determine the label. The basic skip-gram model was then extended to predict not only the context of a word but also the label representing the stock price increase or decrease.

The hypothesis to test can be stated as follows: When the price change label is considered during embedding creation, the distance between embeddings of words occurring more frequently on price decrease and embeddings of words occurring more frequently on price increase will be greater as compared to the embeddings created with the standard Word2Vec skip-gram model. Moreover, this distance should grow with the weight of this price change label represented by u in equation 4.

A. Data and Preprocessing

The experiment utilized tweets related to the Walt Disney Company, which is publicly traded on the New York Stock Exchange under the *DIS* symbol. Tweets published between January 1, 2017 and December 31, 2020 were used to train the Word2Vec model. The *\$DIS* "cashtag" was used to find tweets related to the company. 45 000 tweets containing 401 000 words were further randomly selected in order to reduce both the training time and memory use.

These tweets were preprocessed before they were used as input for the skip-gram model. Following steps were performed:

- the text was transformed to lower-case,
- cashtag symbols \$, hashtag symbols # and the user mention symbols @ were removed,

- all other non-alphanumeric symbols were removed,
- all HTML tags were removed,
- all URLs were replaced by a "__URL__" placeholder,
- and all numbers were replaced by a "__NUMBER__" placeholder.

B. Hypothesis Evaluation

To evaluate the aforementioned hypothesis, the polarity of each word was calculated as the difference between the number of occurrences of a given word in tweets associated with a price increase (positive occurrences) and in tweets associated with a price decrease (negative occurrences).

Since the dataset contains a different number of positive and negative tweets, the polarity of negative tweets was further modified using the following equation:

$$pol_m(w) = \frac{N_{pos}}{N_{neg}} pol(w) \quad (5)$$

where $pol_m(w)$ is the modified polarity of word w , $pol(w)$ is the basic polarity of the same word calculated as the difference between positive and negative occurrences, and N_{pos} and N_{neg} is the total number of tweets related to price increase and price decrease respectively.

Top 75 positive words were then paired with top 75 negative words (most negative word was paired with the most positive word, 2nd most positive word was paired with 2nd most negative word and so on). Then the distances between the embeddings of paired words were examined.

Furthermore, a list of 10 specific antonym pairs was constructed. This list was then verified to make sure that one word in the pair has indeed negative polarity, while the other word's polarity is positive. These antonym pairs are listed in table IV-B. Distances between the embeddings were again examined.

To consider different vector norms, cosine similarity was used as a measure of distance. According to Agudo [1], cosine similarity is also preferred for synonym or antonym detection tasks.

All experiments were implemented in Python 3.9 using well-known libraries including *numpy*, *pandas* and *Keras*.

Positive word	Negative word
buying	selling
upgraded	downgraded
raised	lowered
strength	weakness
ahead	delayed
bullish	bearish
up	down
above	below
high	low
positive	negative

TABLE I
PAIRS OF ANTONYMS WHOSE DISTANCES WERE EXAMINED DURING EXPERIMENTS

V. EXPERIMENT RESULTS

Table II shows the cosine similarity measure for the antonym pairs listed above. These results include three different levels of label weight, as well as similarity derived from

the embeddings trained by the well-known Gensim¹ Word2Vec implementation. With label weight set to 1, the average cosine similarity is very close to the similarity exhibited by Gensim embeddings. This small difference can be attributed to the random nature of the neural network training process. However, as the label weight increases the similarity between embeddings starts to drop significantly. When the label weight is set to 100 the embeddings become almost orthogonal.

These results are confirmed by the second test examining 75 pairs of top positive and top negative words. Noteworthy negative words include "down", "coronavirus", "below", "downgraded" or "risk". Interesting positive words include "higher", "nice", "streaming", "up" or "nflx". Using the same label weight values as in table II, mean cosine similarities across all word pairs were 0.36 ($u = 1$), 0.315 ($u = 10$) and 0.046 ($u = 100$). The cosine similarity mean for Gensim embeddings was 0.346. These results manifest the same behavior as the results for 10 selected antonym pairs.

Both experiments support the hypothesis stated in section IV. Given a specific minimum weight, domain-specific labels can indeed help increase the distance between relevant word embeddings. Moreover, this distance increment grows with the label weight.

VI. CONCLUSION

This paper explored the possible utilization of domain-specific labels during word embedding creation with the Word2Vec algorithm. An extension to the skip-gram model was proposed and evaluated in an experiment where word embeddings were created from a dataset of tweets related to the Walt Disney company. Results of this experiment show that the extension helps distinguish between words whose occurrence is correlated with different labels (in this case stock price increase and decrease). Furthermore, the cosine similarity between such words decreases as the label weight increases.

Presented experiments were performed on a relatively small dataset of 45000 tweets related to a single company. The proposed extension should therefore be examined on significantly larger amounts of data in the future. Moreover, the extension implementation used during experiments is not ready to be deployed to production. Further performance improvements are needed. Combining the extension with negative sampling should also be examined.

The extension was compared with a standard Word2Vec implementation provided by the Gensim library. Additional comparison to the models proposed by [8] or [4] would be beneficial. One of the potential benefits of the presented extension is its ability to consider any domain-specific labels instead of relying on a specific thesaurus.

The work presented in this paper is a part of a larger project with the aim of examining if sentiment and other information

¹<https://radimrehurek.com/gensim/>

TABLE II
COSINE SIMILARITY BETWEEN WORD EMBEDDINGS COMPARED ACROSS VARIOUS CONFIGURATIONS

Positive word	Negative word	Gensim	u = 1	u = 10	u = 100
buying	selling	0.772	0.753	0.156	0.009
upgraded	downgraded	0.961	0.959	0.579	0.082
raised	lowered	0.942	0.937	0.875	0.044
strength	weakness	0.911	0.909	0.594	0.022
ahead	delayed	0.461	0.461	0.547	0.023
bullish	bearish	0.912	0.919	0.151	0.008
up	down	0.602	0.640	0.154	0.004
above	below	0.769	0.735	0.059	0.004
high	low	0.876	0.869	0.117	0.008
positive	negative	0.840	0.844	0.392	0.014
Mean		0.804	0.803	0.362	0.022

extracted from social media can be used to improve mean-risk investment portfolio optimization models. In the future I plan to examine the usefulness of the presented extension for stock price and risk prediction and compare it with complex language models such as BERT.

ACKNOWLEDGMENT

This paper was supported by the SGS project No. SP2022/113. This support is gratefully acknowledged.

REFERENCES

- [1] M. G. Agudo. An analysis of word embedding spaces and regularities. 2019.
- [2] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, and T. Henighan. Language models are few-shot learners. page 25, 2020.
- [3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North*, pages 4171–4186. Association for Computational Linguistics, 2019. doi: 10.18653/v1/N19-1423. URL <http://aclweb.org/anthology/N19-1423>.
- [4] Z. Dou, W. Wei, and X. Wan. Improving word embeddings for antonym detection using thesauri and SentiWordNet. In M. Zhang, V. Ng, D. Zhao, S. Li, and H. Zan, editors, *Natural Language Processing and Chinese Computing*, volume 11109, pages 67–79. Springer International Publishing, 2018. ISBN 978-3-319-99500-7 978-3-319-99501-4. doi: 10.1007/978-3-319-99501-4_6. URL http://link.springer.com/10.1007/978-3-319-99501-4_6. Series Title: Lecture Notes in Computer Science.
- [5] A. Handler. An empirical study of semantic similarity in WordNet and word2vec. page 23, 2014.
- [6] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. 2013. URL <http://arxiv.org/abs/1301.3781>.
- [7] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. 2013. URL <http://arxiv.org/abs/1310.4546>.
- [8] M. Ono, M. Miwa, and Y. Sasaki. Word embedding-based antonym detection using thesauri and distributional information. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 984–989. Association for Computational Linguistics, 2015. doi: 10.3115/v1/N15-1100. URL <http://aclweb.org/anthology/N15-1100>.
- [9] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep contextualized word representations. 2018. URL <http://arxiv.org/abs/1802.05365>.
- [10] I. Samenko, A. Tikhonov, and I. P. Yamshchikov. Intuitive contrasting map for antonym embeddings. 2021. URL <http://arxiv.org/abs/2004.12835>.
- [11] Y. Shao, S. Taylor, N. Marshall, C. Morioka, and Q. Zeng-Treitler. Clinical text classification with word embedding features vs. bag-of-words features. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 2874–2878. IEEE, 2018. ISBN 978-1-5386-5035-6. doi: 10.1109/BigData.2018.8622345. URL <https://ieeexplore.ieee.org/document/8622345/>.
- [12] M. R. Vargas, B. S. L. P. de Lima, and A. G. Evsukoff. Deep learning for stock market prediction from financial news articles. In *2017 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA)*, pages 60–65. IEEE, 2017. ISBN 978-1-5090-4253-1. doi: 10.1109/CIVEMSA.2017.7995302. URL <http://ieeexplore.ieee.org/document/7995302/>.
- [13] H.-Y. Yeh, Y.-C. Yeh, and D.-B. Shen. Word vector models approach to text regression of financial risk prediction. 12(1):89, 2020. ISSN 2073-8994. doi: 10.3390/sym12010089. URL <https://www.mdpi.com/2073-8994/12/1/89>.
- [14] L. Zhang, J. Li, and C. Wang. Automatic synonym extraction using word2vec and spectral clustering. In *2017 36th Chinese Control Conference (CCC)*, pages 5629–5632. IEEE, 2017. ISBN 978-988-15639-3-4. doi: 10.23919/ChiCC.2017.8028251. URL <http://ieeexplore.ieee.org/document/8028251/>.

1st Workshop on Artificial Intelligence for Next-Generation Diagnostic Imaging

THE application of artificial intelligence (AI) becomes nowadays one of the most promising areas of health innovation. Among the most promising clinical applications of AI is diagnostic imaging, where AI algorithms can boost processing power of huge, heterogeneous medical image resources and therefore uncover disease characteristics that fail to be found by the naked eyes. AI can optimize radiologists' workflows and facilitate quantitative assessments of radiographic characteristics using radiomics approach. Nowadays refinement of AI imaging studies is required by consistent selection of clinically meaningful and patient-related endpoints.

The workshop on Artificial Intelligence for Next-Generation Health and Medical Applications – AI4NextGenHMA'22— provides an interdisciplinary forum for researchers, developers, radiologist and clinicians to present and discuss latest advances in research work related to theoretical and practical applications of AI in diagnostic imaging and related areas. The workshop aims to bring together specialists for exchanging ideas and promote fruitful discussions.

TOPICS

The list of topics includes, but is not limited to:

- AI for diagnostic imaging and digital pathology
- DL models and architectures for medical image analysis

- Explainable AI (XAI) for diagnostic imaging
- Radiomics and quantitative image analysis
- Structured reporting in radiology
- Standardization of data annotation for AI
- Clinical decision support systems, especially integrating results of image analysis
- Clinical pathways based on image analysis and integration of clinical data
- Biomedical ontologies, terminologies, standards and clinical guidelines
- Data mining and knowledge discovery in radiology
- Domain knowledge representation and integration into DL/ML models in medical tasks
- Precision medicine and personalized medicine based on diagnostic imaging
- Natural language processing for radiological report analysis
- Medical signal processing and analysis
- Social and ethical aspects of AI in radiology

TECHNICAL SESSION CHAIRS

- **Jóźwiak, Rafał**, Warsaw University of Technology, Poland
- **Pancerz, Krzysztof**, Academy of Zamość, Poland

Development of an AI-based audiogram classification method for patient referral

Michał Kassjański, Marcin Kulawiak
 Department of Geoinformatics,
 Faculty of Electronics, Telecommunications and Informatics,
 Gdansk University of Technology,
 Gdansk, Poland
 Email: {michal.kassjanski, markulaw}@pg.edu.pl

Tomasz Przewoźny
 Department of Otolaryngology,
 Medical University of Gdansk,
 Smoluchowskiego Str. 17,
 80-214 Gdansk, Poland
 Email: tomasz.przewozny@gumed.edu.pl

Abstract—Hearing loss is one of the most significant sensory disabilities. It can have various negative effects on a person’s quality of life, ranging from impeded school and academic performance to total social isolation in severe cases. It is therefore vital that early symptoms of hearing loss are diagnosed quickly and accurately. Audiology tests are commonly performed with the use of tonal audiometry, which measures a patient’s hearing threshold both in air and bone conduction at different frequencies. The graphic result of this test is represented on an audiogram, which is a diagram depicting the values of the patient’s measured hearing thresholds. In the course of the presented work several different artificial neural network models, including MLP, CNN and RNN, have been developed and tested for classification of audiograms into two classes - normal and pathological represented hearing loss. The networks have been trained on a set of 2400 audiograms analysed and classified by professional audiologists. The best classification performance was achieved by the RNN architecture (represented by simple RNN, GRU and LSTM), with the highest out-of-training accuracy being 98% for LSTM. In clinical application, the developed classifier can significantly reduce the workload of audiology specialists by enabling the transfer of tasks related to analysis of hearing test results towards general practitioners. The proposed solution should also noticeably reduce the patient’s average wait time between taking the hearing test and receiving a diagnosis. Further work will concentrate on automating the process of audiogram interpretation for the purpose of diagnosing different types of hearing loss.

I. INTRODUCTION

HEARING IS one of the most important senses and is crucial for a human to maintain full connectivity to the world. Early on in life, hearing helps one to establish language skills which lays the groundwork for quick development during school years. In daily tasks, hearing is used in communicating with other people as well as for listening to music, television and radio, and going to the cinema or theatre.

According to World Health Organization (WHO), currently, around 430 million people globally require rehabilitation services for their hearing loss [1]. Estimations show that by 2050 nearly 2.5 billion people will be living with some degree of hearing loss, at least 700 million of whom will require rehabilitation services [1]. Overall, hearing impairment has devastating consequences for interpersonal communication, psychosocial well-being, quality of life and economic inde-

pendence [2]. The consequences of hearing loss are frequently underestimated and ignoring the initial symptoms usually leads to further degradation. Once diagnosed, early intervention is the key to successful treatment. Medical and surgical treatment can cure most ear diseases, potentially reversing the associated hearing loss. Research has shown that, particularly in children, almost 60% of hearing loss is due to causes that can be prevented [1], [6], [7].

The standard hearing test is carried out using pure tone audiometry, which determines the hearing thresholds at different frequencies. As a rule, a frequency range of the hearing test varies within 125 – 8000 Hz. The sound level of pure tones is given in dBHL, and the subject is tested in both air and bone conduction. The test results in two data series containing discrete hearing thresholds in the function of frequency, separately for both conductions. This data series is usually presented in the form of an inverted graph called audiogram. An audiogram helps to determine the degree of hearing loss, but also the type of pathology: sensorineural, conductive or mixed [3], [4].

According to projections, the demand for professional audiologists will burgeon in near future [1]. Nowadays, around 78% of low-income countries have less than one otorhinolaryngologist per million inhabitants and about 93% have less than one audiologist per million inhabitants [1], [5]. In this context, introduction of expert systems based on artificial intelligence for preliminary audiogram interpretation could significantly reduce the workload of specialists, while at the same time shortening the patient’s wait for a diagnosis.

Over the last decade, a comparison of several approaches to hearing loss determination, including Decision Tree, Naive Bayes and Neural Network Multilayer Perceptron (NN) model, has been prepared by Elbaşı & Obalı [10]. The tests have been carried out using a set of numerical values representing Decibels corresponding to fixed frequency levels (750Hz, 1kHz, 1.5kHz, 2kHz, 3kHz, 4kHz, 6kHz, 8kHz). The achieved accuracy was 95.5% in Decision Tree, 86.5 % in Naive Bayes and 93.5 % in NN.

A different approach was presented by Noma & Ghani [11], who developed a classification system based on the relationship between pure-tone audiometry thresholds and inner ear

disorders symptoms such as Tinnitus, Vertigo, Giddiness etc. The classifier, based on the multivariate Bernoulli model with feature transformation, has shown to provide 98% accuracy of predicting hearing loss symptoms based on audiometry results.

Recently, Charih et al. [12] presented their Data-Driven Annotation Engine, a decision tree based audiogram classifier which considers the configuration, severity, and symmetry of participant's hearing losses and compared it to AMCLASS [13], which fulfils the same purpose using a set of general rules. Both classifiers have achieved similar accuracy of around 90% across 270 different audiometric configurations by three licensed audiologists.

More recently, Crowson et al. [14] adopted the ResNet-101 model to classify audiogram images into three types of hearing loss (sensorineural, conductive or mixed) as well as normal hearing using a set of training and testing images consisting of 1007 audiograms. This approach resulted in 97.5% classification accuracy, however it is limited to processing images.

In summary, the combination of neural networks and increased computing resources of new hardware architectures has the potential to deliver faster overall tests results and more detailed assessments[15]. This being said, however, the currently proposed solutions deliver classification accuracy in the 90-95% range, which, although very high, still leaves considerable room for error. Clinical standards suggest that the margin of error should be kept under 5%[16] and optimally should be close to 3% [17]. These requirements are met only by two of the discussed classifiers. The method proposed by Noma & Ghani achieves 98% accuracy, however it has been designed to predict significant symptoms of inner ear disorder, and thus it cannot be used for general purposes such as early detection of hearing degradation. The best audiogram classifier to date has been presented by Crowson et al., who used transfer learning to adapt an established image classifier network to analysis of audiogram images. While this approach resulted in a 97% classification accuracy, it exhibits serious limitations. Because it is an image classifier, it cannot be used with the original data series produced by tonal audiometry. This means that the data series first need to be converted into audiogram images, which may result in data loss. Moreover, although the structure of audiograms generally is similar, there can still be significant differences between audiograms generated by different hardware and software configurations. Aside from differences such as background and line colours, audiograms can also differ in the amount of presented information (eg. they may contain data for a single ear or both). A sample comparison of significant differences between audiograms obtained from different sources is presented in Figures 1 and 2. In consequence, a universal solution for classifying results of tonal audiometry cannot be based on an image classifier.

This study presents the development of a neural network for classification of discrete tonal audiometry data series. In the course of this study, several different neural network architectures have been trained and tested with the use of 2400 audiogram data series analysed and classified by professional audiologists. The goal of the presented study was to achieve a

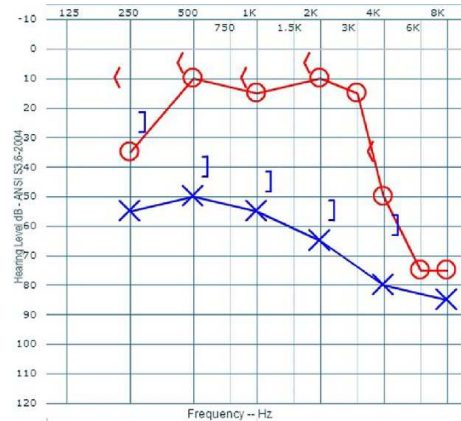


Fig. 1. A pure tone audiogram showing air and bone conduction thresholds for **both left and right ear** [8]. The "X" and "O" symbols are used to mark left-sided air and bone conduction, respectively. The "O" indicate air conduction, whereas the "<" denote bone conduction, both in the right ear.

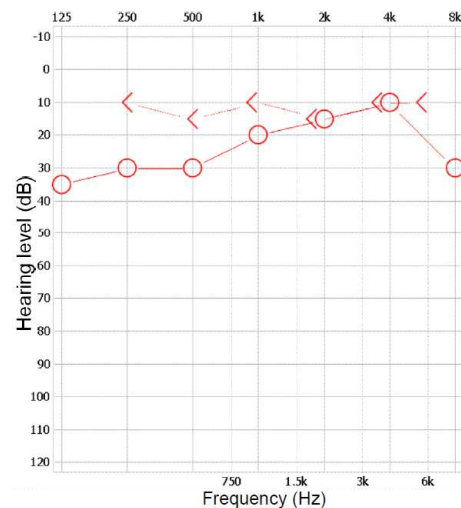


Fig. 2. A pure tone audiogram showing air and bone conduction thresholds **only for the right ear**. The "O" and "<" indicate left-sided air and bone conduction, respectively.

high enough classification accuracy for the developed network to be applicable for use in a clinical environment.

II. MATERIALS & METHODS

A. Data

The study has been conducted with the use of 2400 data series containing results of pure tone audiometry tests performed from 2020 to 2021 by clinicians working at the Otolaryngology Clinic of the University Clinical Centre in Gdansk, Poland. The data contains 650 examples of normal hearing and 1750 examples of pathological hearing loss. The tests had been performed in a soundproof booth, according to ISO 8253 and ISO 8253 standards. Air conduction tests employed TDH-39P headphones, while bone conduction testing involved a Radioear B-71 bone-conduction vibrator. The data series have been analysed and labelled by expert audiologists from the

Medical University of Gdansk Department of Otolaryngology according to established methodology [9]. In consequence, the dataset has been classified into two subsets: hearing pathology and normal hearing.

B. Preprocessing

The input data series contained numerical information about tonal points, defined as loudness (dB) for a given frequency (Hz), in XML format. The dataset included the following range of frequencies:

125Hz, 250Hz, 375Hz, 500Hz, 750Hz, 1000Hz, 1500Hz, 2000Hz, 3000Hz, 4000Hz, 6000Hz, 8000Hz.

Every tested frequency has been assigned a loudness level in the range from -10dB to 120dB. If certain frequencies had not been registered during the hearing test, they have not been included in the corresponding data series.

C. Testing methodology

Using the prepared dataset, three different neural network architectures have been trained to interpret tonal audiometry data and in order to differentiate normal hearing (N) from pathological hearing loss (P). The tested architectures included Multilayer Perceptron (MLP), Convolutional (CNN) and Recurrent (RNN) neural networks, all of which have been previously applied to data classification problems [18], [19], [20]. The general workflow of the presented study is shown in Fig. 3. Each model has been assessed using k-fold cross-validation, which consists of dividing the data into k subsets and training the model k-times with k-1 subsets, with a different subset being used for testing in every iteration. The presented research used $k = 5$, which resulted in train to test dataset proportions of 80% to 20%, respectively.

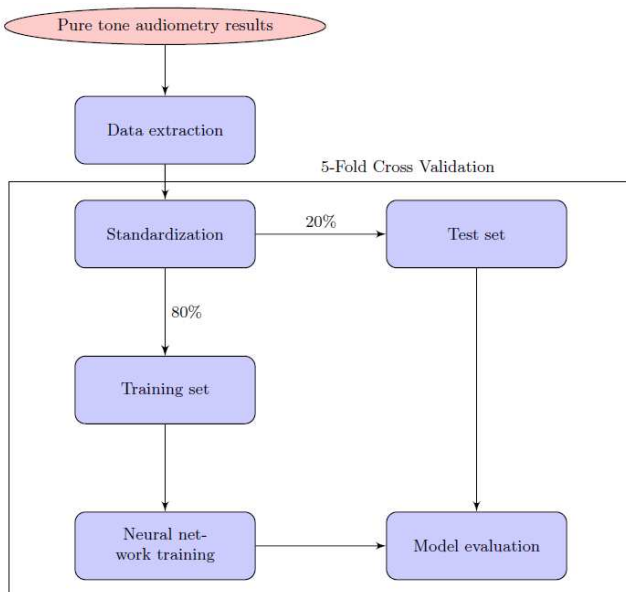


Fig. 3. Workflow of processes leading to model evaluation.

After revealing the best performing architecture, further tests and optimizations would be carried out in order to improve classification accuracy.

III. RESULTS

The purpose of the initial tests was to reveal the best neural network architecture model for classification of pure tone audiometry data. The tested neural network architectures included MLP, CNN and RNN. The results of those tests are presented in Table I.

TABLE I
COMPARISON OF PERFORMANCE RESULTS OF PRELIMINARY MODELS.

Parameters	MLP	CNN	RNN
Accuracy	0.9458	0.9563	0.9604
Loss	0.6429	0.1185	0.1346
Precision	0.8255	0.8984	0.9062
Recall	1.0	0.9349	0.9430
F1	0.9044	0.9163	0.9243

As it can be seen, initial research revealed that the best classification performance has been produced by the RNN architecture model. Once the most promising neural network architecture has been identified, three of its variants have been trained and optimized in terms of hyper parameters, including number of nodes and hidden layers, dropout layers, learning and decay rate. The first model consisted of a simple RNN, second one was based on Gated Recurrent Units (GRU) [22] and the last one used Long Short-Term Memory (LSTM) [21]. The results of these tests are shown in Table II.

Receiver Operating Characteristics (ROC) curves with corresponding Area Under the Curve (AUC) parameters for these models are presented in Fig. 4.

TABLE II
COMPARISON OF PERFORMANCE RESULTS OF RNN MODELS.

Parameters	Simple RNN	GRU	LSTM
Accuracy	0.9646	0.9771	0.9812
Loss	0.0836	0.0530	0.0540
Precision	0.9030	0.9453	0.9394
Recall	0.9680	0.9680	0.9920
F1	0.9344	0.9565	0.9650

The cross validation scores for $k = 5$ with LSTM classifier are given in Table III. The average accuracy was 98.08% (+/- 0.17%).

TABLE III
K-FOLD VALIDATION SCORE OF LSTM MODEL ($k = 5$).

Iteration	1	2	3	4	5
Accuracy	97.96	98.33	97.96	97.91	98.22

A detailed analysis of classification performance achieved by the tested RNN models can be made using a confusion matrix, which visualizes the number of True Positives (TP

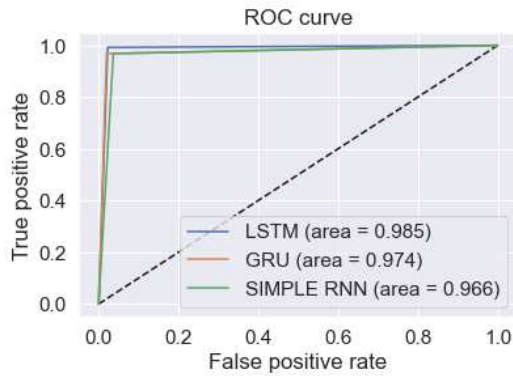


Fig. 4. ROC curve with AUC parameter of RNN models.

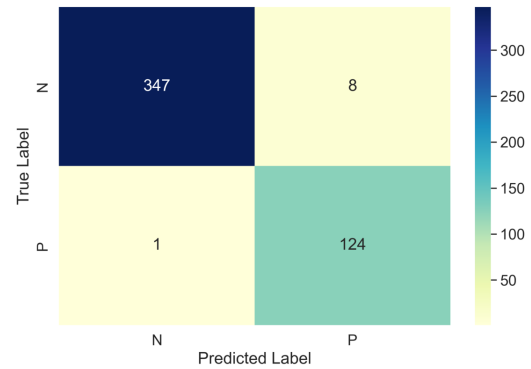


Fig. 7. Confusion matrix of LSTM.

- patients who have been properly classified with hearing loss), True Negatives (TN - patients who have been properly classified with good hearing), False Positives (FP - patients who have been improperly classified as hearing loss) and False Negatives (FN - patients who have been improperly classified with good hearing). The confusion matrix for the tested RNN models is presented in Figures 5, 6 and 7.

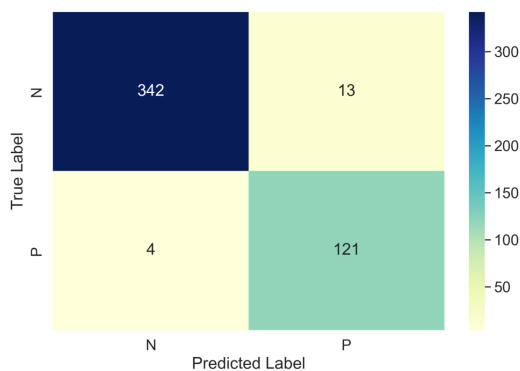


Fig. 5. Confusion matrix of simple RNN.

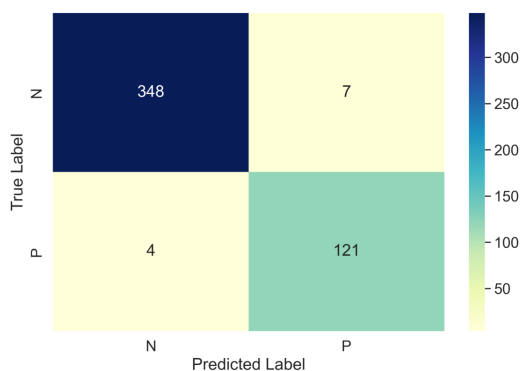


Fig. 6. Confusion matrix of GRU.

IV. DISCUSSION

Initial tests have shown that the simple RNN architecture model delivers noticeably better pure tone audiometry classification results in comparison to MLP and CNN models, achieving accuracy of 96.04% versus 94.58% and 95.63% respectively (Tab. I). The chosen network architecture appears to have the largest impact on classification accuracy, as further tests and optimizations resulted in minor improvements. Optimization of parameters such as the number of nodes and hidden layers, dropout layers as well as learning and decay rate improved the accuracy of simple RNN from 96.04% to 96.46%. In comparison, applying the same optimization process to MLP and CNN models did not result in markedly improved evaluation parameters. A possible explanation for this could be the fact that RNN have been designed to process time series data, and structurally pure tone audiometry results could be interpreted as a special case of time series. This could be further explored by testing the effectiveness of more advanced RNN models such as GRU and LSTM. As it can be seen in Tab. II, both of these models obtained more than 97% accuracy, with the highest out-of-training set accuracy being achieved by LSTM at 98.12%. While these results, which have been cross-validated using the 5-fold method, would seem to indicate a general prevalence of the RNN architecture in processing audiometry data, establishing an effectiveness hierarchy of RNN models is a more complex matter. Although LSTM has shown the best classification accuracy, when analysed in terms of confusion matrix, the lowest number of False Positives (FP) was obtained by GRU (Figures 6 and 7), with LSTM taking second place. In comparison, the simple RNN produced over 62% more False Positives than LSTM and 85% more than GRU.

Overall, simple RNN and GRU performed equally well in terms of False Negatives (FN), producing them only in 0.8% of cases, whereas LSTM significantly outperformed the other models with only one case of error occurring. It can be argued that when classifying results of pure tone audiometry tests, the FN number is more important than FP because it shows that a patient does not have hearing loss when they actually do. In this case the patient may not receive treatment and

get worse because their disease was undetected. On the other hand, a false positive would only result in the patient being unnecessarily referred to an audiologist, who would properly interpret the test results and inform the patient that their level of hearing is normal.

Summing up, it can be said that the 98.12% classification accuracy achieved by LSTM fulfills the established margin of error criteria and is significantly better than the 97.5% classification accuracy offered by the best existing algorithm for audiogram data classification, proposed by Crowson et al. [14]. While some of the difference could be attributed to the rival method providing a larger set of classes, the presented method provides an additional advantage in the type of processed data: it works with original tonal audiometry data series instead of audiogram images and therefore is more universal. The only rival method also designed for processing tonal audiometry data series, presented by Elbaşı & Obali [10], provides an even lower 95.5% classification accuracy.

In terms of classifying pure tone audiometry data, the only existing solution with a similar classification accuracy level (98%, proposed by Noma & Ghani [11]), has been designed to predict significant symptoms of inner ear disorder and thus cannot be used for general classification of tonal audiometry test results.

V. CONCLUSIONS

The presented work aimed to develop a neural network for classification of discrete tonal audiometry data series with accuracy high enough for medical application. In the course of this study, several different neural network architectures, including MLP, CNN and RNN, have been trained and tested with the use of 2400 audiogram data series analysed and classified by professional audiologists. The highest classification accuracy was achieved with an optimized LSTM RNN at 98.12%. The high accuracy of the obtained neural network, particularly the low number of False Negatives (0.2%), allows for its application at the Otolaryngology Clinic of the University Clinical Centre in Gdansk, Poland. Results of pure tone audiometry tests, which thus far needed to be examined by professional audiologists, can now be classified with the developed neural network under the supervision of general practitioners. This change may result in a significant reduction of the workload of audiology specialists, as they will no longer need to deal with patients whose symptoms are not caused by hearing loss (which may amount to over 10% of all patients subjected to pure tone audiometry tests) [23], [24]. After it has been further tested in practice, the developed solution could be introduced directly in the audiometry laboratory, ensuring that the patient receives a first interpretation of the performed tests as soon as they have been completed. Further work will concentrate on expanding the classifier for the purpose of diagnosing different types of hearing loss.

ACKNOWLEDGEMENT

The authors would like to thank M. Grono, K. Koźmiński, P. Mierzwińska and A. Romanowicz who helped to create the

pure tone audiometry test dataset used in this study.

REFERENCES

- [1] World Health Organization. 2021. World report on hearing. <https://www.who.int/publications/i/item/world-report-on-hearing>.
- [2] Olusanya, B. O., Neumann, K. J., Saunders, J. E. 2014. The global burden of disabling hearing impairment: a call to action. *Bull World Health Organ.* 92(5):367–373, <http://dx.doi.org/92/5/13-128728>
- [3] Kapul AA, Zubova EI, Torgaev SN, Drobchik VV. 2017. Pure-tone audiometer. *J Phys Conf Ser*, <http://dx.doi.org/10.1088/1742-6596/881/1/012010>
- [4] V. P. Aras. 2003. Audiometry techniques, circuits, and systems, M. Tech. Credit Seminar Report, Electronic Systems Group, EE Dept, IIT Bombay
- [5] World Health Organization. 2013. Multi-country assessment of national capacity to provide hearing care
- [6] Tukaj C, Kuczkowski J, Sakowicz-Burkiewicz M, Gulida G, Tretiakow D, Mionskowski T, Pawełczyk T. 2014. Morphological alterations in the tympanic membrane affected by tympanosclerosis: ultrastructural study. *Ultrastruct Pathol.* 38(2):69-73, <http://dx.doi.org/10.3109/01913123.2013.833563>
- [7] Narozny W, Skorek A, Tretiakow D. 2021. Does Treatment of Sudden Sensorineural Hearing Loss in Patients With COVID-19 Require Anticoagulants? *Otolaryngol Head Neck Surg.* 165(1):236-237, <http://dx.doi.org/10.1177/0194599820988511>
- [8] Prashanth Prabhu P, Jyothi Shivswamy. 2017. Audiological findings from an adult with thin cochlear nerves. *Intractable & Rare Diseases Research*, 6(1):72-75, <http://dx.doi.org/10.5582/irdr.2016.01081>
- [9] Przewoźny T, Kuczkowski J. 2017. Hearing loss in patients with extracranial complications of chronic otitis media. *Otolaryngol Pol.* 71(3), pp. 31-41, <http://dx.doi.org/10.5604/01.3001.0010.0130>
- [10] Ersin Elbaşı, Murat Obali. 2012. Classification of Hearing Losses Determined through the Use of Audiometry using Data Mining. Conference: 9th International Conference on Electronics, Computer and Computation
- [11] Noma, N. G., & Ghani, M. K. A. 2013. Predicting Hearing Loss Symptoms from Audiometry Data Using Machine Learning Algorithms. In *Proceedings of the Software Engineering Postgraduates Workshop (SEPoW)*, p. 86, Penang, Malaysia
- [12] Charih F, Bromwich M, Mark AE, Lefrançois R, Green JR. 2020. Data-Driven Audiogram Classification for Mobile Audiometry. *Sci Rep* 10, 3962, <http://dx.doi.org/10.1038/s41598-020-60898-3>
- [13] Margolis, R.H. and Saly, G.L. 2007. Toward a standard description of hearing loss. *International journal of audiology*, 46(12), pp.746-758, <http://dx.doi.org/10.1080/14992020701572652>
- [14] Crowson MG, Lee JW, Hamour A, Mahmood R, Babier A, Lin V, Tucci DL, Chan TCY. 2020. AutoAudio: Deep Learning for Automatic Audiogram Interpretation. *J Med Syst.* 44(9):163, <http://dx.doi.org/10.1007/s10916-020-01627-1>
- [15] Barbour, Dennis L. MD, PhD; Wasmann, Jan-Willem A. 2021. Performance and Potential of Machine Learning Audiometry, *The Hearing Journal: Volume 74 - Issue 3 - p 40,43,44*, <http://dx.doi.org/10.1097/01.HJ.0000737592.24476.88>
- [16] Aziz, B., Riaz, N., Rehman, A.U., Malik, M.I., Malik, K.I. 2021. Colligation of Hearing Loss and Chronic Otitis Media. *Pakistan Journal of Medical and Health Sciences* Vol. 15, Issue 8, pp. 1817, <http://dx.doi.org/10.53350/pjmhs211581817>
- [17] Raghavan, A., Patnaik, U. and Bhaudaria, A.S. 2020. An Observational Study to Compare Prevalence and Demography of Sensorineural Hearing Loss Among Military Personnel and Civilian Population. *Indian Journal of Otolaryngology and Head & Neck Surgery*, pp.1-6, <http://dx.doi.org/10.1007/s12070-020-02180-6>
- [18] Zieliński, S. K., & Lee, H. 2018. Feature extraction of binaural recordings for acoustic scene classification. In 2018 Federated Conference on Computer Science and Information Systems (FedCSIS) (pp. 585-588), <http://dx.doi.org/10.15439/2018F182>
- [19] Agbehadji, I. E., Millham, R., Fong, S. J., & Yang, H. 2018. Kestrel-based Search Algorithm (KSA) for parameter tuning unto Long Short Term Memory (LSTM) Network for feature selection in classification of high-dimensional bioinformatics datasets. In 2018 Federated Conference on Computer Science and Information Systems (FedCSIS) (pp. 15-20), <http://dx.doi.org/10.15439/2018F52>

- [20] Lindén, J., Forsström, S., & Zhang, T. 2018. Evaluating combinations of classification algorithms and paragraph vectors for news article classification. In 2018 Federated Conference on Computer Science and Information Systems (FedCSIS) (pp. 489-495), <http://dx.doi.org/10.15439/2018F110>
- [21] Hochreiter, Sepp & Schmidhuber, Jurgen. 1997. Long Short-term Memory. *Neural computation*. 9. 1735-80 (1997), <http://dx.doi.org/10.1162/neco.1997.9.8.1735>
- [22] Cho, Kyunghyun & Merriënboer, Bart & Gulcehre, Caglar & Bougares, Fethi & Schwenk, Holger & Bengio, Y. 2014. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation, <http://dx.doi.org/10.3115/v1/D14-1179>
- [23] do Carmo LC, Médicis da Silveira JA, Marone SA, D'Ottaviano FG, Zagati LL, Dias von Söhsten Lins EM. 2018. Audiological study of an elderly Brazilian population. *Braz J Otorhinolaryngol*;74(3):342-9, [http://dx.doi.org/10.1016/s1808-8694\(15\)30566-8](http://dx.doi.org/10.1016/s1808-8694(15)30566-8)
- [24] Walker JJ, Cleveland LM, Davis JL, Seales JS. 2013. Audiometry screening and interpretation. *Am Fam Physician*.;87(1):41-7

Canine age classification using Deep Learning as a step toward preventive medicine in animals

Szymon Mazurek [1,2], Maciej Wielgosz [1,2], Jakub Caputa [1], Rafał Frączek [1,2], Michał Karwatowski [1,2],
 Jakub Grzeszczyk [1], Daria Łukasik [1], Anna Śmiech [3], Paweł Russek [1,2], Ernest Jamro [1,2],
 Agnieszka Dąbrowska-Boruch [1,2], Marcin Pietroń [1,2], Sebastian Koryciak [1,2], Kazimierz Wiatr [1,2]

1 ACC Cyfronet AGH, Nawojki 11, 30-950 Kraków, Poland

2 AGH University of Science and Technology, al. Mickiewicza 30, 30-059 Kraków, Poland

3 University of Life Sciences, al. Akademicka 13, 20-950 Lublin, Poland

Email: {rafalfr, mkarwat, wielgosz, russek, adabrow, jamro, pietron, koryciak, wiatr}@agh.edu.pl

{s.mazurek, d.lukasik, j.grzeszczyk, j.caputa}@cyfronet.pl

anna.smiech@up.lublin.pl

Abstract—The main goal of this work was to implement a reliable machine learning algorithm that can classify a dog’s age given only a photograph of its face. The problem, which seems simple for humans, presents itself as very difficult for the machine learning algorithms due to differences in facial features among the dog population. As convolutional neural networks (CNNs) performed poorly in this problem, the authors took another approach of creating novel architecture consisting of a combination of CNN and vision transformer (ViT) and examining the age of the dogs separately for every breed. Authors achieved better results than those in initial works covering the problem.

I. INTRODUCTION

DELAYING the aging and preventing diseases associated with it is becoming increasingly prevalent in state-of-the-art medicine. Biological and chronological age comparison is one of the primary assessment tools to estimate the health status of a given subject. The trend is also prevalent in veterinary medicine as animals, especially domesticated ones such as dogs, play an essential role in modern societies.

Machine learning and its growing effectiveness and versatility provide valuable tools that can be used in medicine and anti-aging research to create simple and easy-to-use methods for assessing biological age.

The age assessment can help approximate an animal’s health status, allowing the animal owner to take preventive actions before a health condition is developed. The authors hope this work’s findings will also be useful for further research in human and animal preventive medicine.

The main goal of this work was to create a tool to assess a dog’s age, given only its picture. The age groups of the dogs were classified as follows:

- Young dogs, aged from 0 to 2 years,
- Adult dogs, aged from 2 to 6 years,
- Senior dogs, aged from 6 years and more.

The tool will be a part of the larger project aiming to detect animals’ health conditions and belongs to a realm of preventive medicine solutions. The system’s overall architecture is presented in Fig. 1.

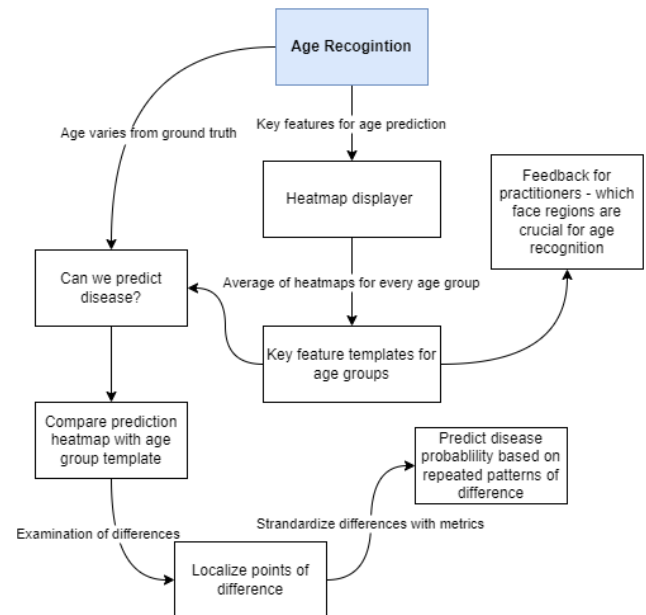


Fig. 1. A block diagram of the CyfroVet system for early disease diagnostics (a part of it presented in [11])

This work is part of CyfroVet, a research project run by ACC Cyfronet to utilize technology, especially AI, in veterinary medicine. The anti-aging and presented system is one of the topics covered by the project, alongside cancer cell detection and segmentation.

The summary of this paper’s contributions is as follows:

- new architecture for addressing the difficult task of dogs classification, which outperforms available state-of-the-art models,
- new dataset which enables effective training of the classification models,
- conceptualization of using age assessment in animals for disease prevention and their early diagnosis.

We make our code and the dataset publicly available.

II. MATERIALS AND METHODS

There are multiple methods for assessing canines age. Most of them involve a certain kind of precise examination. The few most popular are:

- Ocular lens examination [1]
- Teeth examination [2]
- Bone examination [3]

Authors have noticed an opportunity to use computer vision in this field, as it can recognize image features and patterns sometimes impossible to notice for humans with a bare eye examination. As mentioned before, a reliable age assessment based on appearance can be a straightforward tool to use for veterinary medicine practitioners and pet owners. Facial appearance differing vastly from the typical one for a given age group can be a sign of accelerated biological aging, which may be an indicator of the overall health status of an individual. Having preliminary warning signs, practitioners can take preventive steps and make a detailed diagnosis to detect early signs of disease before more severe symptoms develop. This early warning can be beneficial in treating potentially terminal conditions in their early stage. Quick detection and treatment of medical conditions such as cancer can significantly increase an individual's chances of survival.

A. Datasets

In this experiment, the authors initially decided to use the DogAge dataset containing dogs of various age groups created by the Tech4Animals research group [5]. This dataset was used in the Dog Age Challenge competition organized by the group mentioned earlier, which required participants to create a neural network algorithm that could assess the dog's age. It took place in 2019.

The expert dataset consisted of photographs classified into age groups by animal scientists. The photographs were high-quality, and the dog was directly facing the camera.

The Petfinder dataset consisted of photographs downloaded from *Petfinder.com*, a website for pet adoption [4]. Experts did not verify the classification of each age group. The images were often of poor quality, with dogs at various angles and distances.

Several preprocessing techniques were used for those datasets, such as cropping and data augmentation. They will be described in detail in the next section. The Petfinder dataset was cleared of irrelevant photos (i.e., dog's face not clearly visible, other dogs appearing).

After the initial experiments, the authors sought to expand the datasets, as the performance was not reaching satisfying results. The data was collected from the *Petfinder.com* [4], as the website provided the divisions of the dogs by age and breed. The photographs were downloaded and cleared, as many repeating adoption offers were on the website. Two approaches were taken, with the first dividing the problem into examining only one breed at a time and the second by expanding the mixed breeds datasets. Obtained photos were similar to those described in the previous case of the

Petfinder dataset. The datasets are described in Tab. I. *Expert dataset* and *Petfinder dataset* refer to datasets presented in DogAge Challenge. *Big dataset* refers to a dataset created by downloading new photos. Two datasets consisting only of dogs of single breeds are also presented in Tab. I. The dataset name describes the breeds. Given breeds were selected based on data availability - the number of photos and class balance.

III. EXPERIMENTS

The initial datasets presented many challenges. Firstly, they were greatly imbalanced, with the adult category containing more images than the other two combined. Secondly, the data quality was also an obstacle - the *Petfinder* set contained many invalid photos (i.e., dog's face not visible, humans present in the image) for the network training. Those problems were also present within the new datasets. The experiments were run on the Prometheus cluster, using one node with 2 Nvidia Tesla v100 GPUs and 36 CPU cores.

A. Image cropping algorithm

A two-way approach was introduced to address the problem of low image quality. Firstly, the *Petfinder* dataset was manually cleared, removing the inadequate pictures (e.g., a dog looking back or to the side, two dogs in the picture, dog with toys in its mouth). Secondly, two cropping algorithms were tested: YOLO (You Only Look Once)[6] and FaceDetector. After the evaluation, the latter was chosen. Both algorithms could correctly detect the dogs. However, FaceDetector focused only on the dog's face. Therefore, it presented itself not only as a cropping mechanism but also as a tool to clear the new datasets of irrelevant photos, as manual clearing is time inefficient. Results of both cropping algorithms are visible in Fig. 2. for FaceDetector and Fig. 3. for YOLO.



Fig. 2. An image cropped with FaceDetector



Fig. 3. An image cropped with YOLO

B. Metrics

Categorical accuracy (CA), F1-score (F1), and recall (RA) metrics were used to examine the performance of tested models. In the DogAgeChallenge announcement, [10] a modified version of mean average error (MAE), average accuracy (ACA), and average recall (aRA) is used. In this paper, the authors decided not to use mMAE since it was concluded that this metric would provide no meaningful information.

TABLE I
SUMMARY OF THE DATASETS

Dataset name	Total number of photos	Number of adult group photos	Number of senior group photos	Number of young group photos
Expert dataset	1088	370	495	223
Petfinder dataset	22573	13016	1898	7659
Big dataset	49010	26021	5523	17464
Big dataset (balanced)	16964	5999	5517	5448
Chihuahua	3933	1898	875	1160
Pitbull terrier (balanced)	3195	1128	996	1071

C. Augmentation, class weights and data balancing

To cope with a class imbalance within the datasets, the authors tried to introduce data augmentation via random flip, rotation, brightness, and contrast changes. However, augmentation harmed models' performances, lowering CA on validation sets on average by 5% when augmenting by flip and rotation and 10% when augmenting via contrast and brightness changes. Those tests were run on a balanced Pitbull Terrier dataset, excluding the influences of class imbalance being magnified by augmentation. Authors suspect that augmentation via brightness and contrast shifts may negatively influence the results by skewing some features crucial for age assessment (for example, fur color). Data augmentation failed to solve the class imbalance problem, so the authors used class weights during training. Weights were chosen as 1, 5, and 2 for Adult, Senior and Young classes. Only the Expert dataset, showing different data distribution, was assigned weights 2, 1, and 2 for the same class order. Also, class balancing via removal of the photos from most numerous classes was tested. The number of removed photos was chosen so that all classes had roughly the same number of photos as the least numerous ones. In the case of balanced datasets, no class weights were applied in the final experiments, as this approach did not perform well for balanced datasets (once again reducing CA and F1).

D. Initial network architecture and experiments

Initial trials were conducted using the transfer learning protocol. The backbone of the model consisted of CNN pre-trained on ImageNet dataset with a classifier consisting of fully connected layers. During the experiments, the EfficientNetB7 [7] was chosen as the final backbone for the model, as the results of different commonly used CNN architectures were performing on a very similar level. The architecture of the initial model is based on the EfficientNetB7 backbone, followed by global average pooling and two fully connected layers.

The initial model was tested on the available datasets. During the experiments, the random data splits were as follows:

- training split - 80% of the dataset
- validation split - 10% of the dataset
- test split - 10% of the dataset

The network was tested on both single and mixed-breed datasets. It was trained using AdamW [8] optimizer with a learning rate parameter equal to 0.0001 and the weight decay

TABLE II
ARCHITECTURE OF THE NETWORK WITH ViT AS CLASSIFIER

Layer number	Layer name	Parameters
1	EfficientNetB7	Weights from pretraining on Imagenet
2	Patch Encoding 2D	Embedding dimension of 64
3	Layer normalization	-
4	Multi-Head Attention	6 attention heads
5	Skip connection	Adding outputs from layers 2 and 4
6	MLP layer	-
7	Skip connection	Adding outputs from layer 5 and 6
8	Layer normalization	-
9	Flattening layer	-
10	Dropout layer	50% dropout
11	MLP layer	dropout increased to 50%
12	Dense layer	Softmax activation

parameter of the same value. Different configurations of the classifier with the usage of regularization, dropout layers, more dense layers, and different values of units within dense layers did not significantly influence the model's performance. The results of experiments with the initial network are visible in Tab. III and Tab. IV.

E. Transformer-based classifier as a method to improve the classification results

During the experiments, the CNN model obtained better results than those found in the literature [10]. However, they still were not satisfying, so the authors decided to modify the architecture, adding a small version of ViT as a classifier after the backbone CNN block. ViT implements an attention-based model known in the field of NLP. The decoder block is omitted in the computer vision version compared to the original implementation. The model uses an attention mechanism to learn relations between different parts of the input image. Dosovitskiy et al. [9] showed that ViT can outperform current state-of-the-art CNN architectures in image classification. As it requires large datasets, the authors decided to extract features from the data using pre-trained CNN. Architecture is presented in Tab. II.

The model was again trained using available datasets, with the same data split as for experiments with the initial network. Results are visible in Tab. III and Tab. IV.

TABLE III

FINAL RESULTS OF THE INITIAL AND MODIFIED NETWORKS TRAINED ON THE DESCRIBED DATASETS.

Model	Initial network		Network with ViT as classifier	
	CA	RA	CA	RA
Expert dataset	0.49	0.44	0.51	0.49
Petfinder dataset	0.49	0.23	0.55	0.50
Big dataset	0.44	0.23	0.46	0.29
Big dataset (balanced)	0.53	0.25	0.54	0.45
Chihuahua	0.45	0.35	0.48	0.45
Pitbull Terrier (balanced)	0.50	0.22	0.50	0.44

TABLE IV

FINAL F1 SCORE OF THE INITIAL AND MODIFIED NETWORKS TRAINED ON THE DESCRIBED DATASETS FOR SPECIFIC CLASSES (A - ADULT, S - SENIOR, Y - YOUNG).

Model	Initial network			Network with ViT as classifier		
	A	S	Y	A	S	Y
Expert dataset	0.47	0.57	0.35	0.33	0.64	0.41
Petfinder dataset	0.47	0.40	0.55	0.57	0.42	0.56
Big dataset	0.28	0.42	0.56	0.32	0.45	0.57
Big dataset (balanced)	0.45	0.61	0.53	0.49	0.60	0.55
Chihuahua	0.27	0.54	0.55	0.49	0.51	0.43
Pitbull Terrier (balanced)	0.43	0.60	0.48	0.55	0.40	0.50

F. Results

The results obtained during the experiments are presented in this section compared to the previous experiments. In the comparison, classification accuracy (CA) and recall (RA) were used to compare the obtained results with the ones conducted previously during the DogAge Challenge [10]. During DogAge Challenge, authors showed that using Squeezenet and Inception v3 with dense layer classifiers resulted in the CA and RA of 32% for the Squeezenet model and 34% on both metrics for Inception v3. Comparing these values with the ones seen in Tab. III, which describes the ones obtained with solutions proposed by the authors trained on available datasets, we can see that significant improvement is made. The networks can reach both higher CA and RA. The CNN + ViT architecture improved CA and RA for every dataset. The single breed approach was similarly effective compared to the mixed breed approach. Examining Tab. IV, it is visible that CNN + ViT does not necessarily improve F1 for given classes but instead results in more even per-class F1. It can also be concluded that the class balancing and weighting approach can be a helpful tool to cope with class imbalances for this problem, as balanced datasets' results are comparable to those obtained on unbalanced datasets with class weights applied during network training.

IV. DISCUSSION

Several challenges affect the experiment and results presented in this work. The most important of them are as follows:

- limited access to high-quality data,

- lack of the effective data specification methods, which makes both augmentation and preparation of the validation dataset challenging,
- it is tough to deal with multiple breeds using a single model,
- granularity of the age classification is tightly related to an ability to label images precisely.

Other research projects in this domain may also consider these remarks to reach a high performance of ML models.

V. CONCLUSIONS AND FUTURE WORK

The problem of dog age classification using neural networks turned out to be more demanding than expected in the beginning. Experiments showed that state-of-the-art CNN models provided insufficient accuracy while using datasets with multiple dog breeds. The transformer-based architecture introduced in the paper improved the performance results, but there is still much space for progress in future research.

VI. DATA AND CODE AVAILABILITY

The code and datasets as well as additional results available: github.com/SzymonMazurekAGH/Age_recognition_Cyfrovet

REFERENCES

- [1] Abood S. Estimating Age in Dogs and Cats Using Ocular Lens Examination. *Compendium on Continuing Education for the Practising Veterinarian - North American Edition*. (2000).
- [2] Roccaro, M. & Peli, A. A determination in dog puppies by teeth examination: legal, health and welfare implications, review of the literature and practical considerations. *Veterinaria Italiana* 56. pp. 149-162 (2020). DOI: 10.12834/VetIt.1876.9968.2
- [3] Sutton, L., Byrd, J. & Brooks, J. Age Determination in Dogs and Cats. *Veterinary Forensic Pathology, Volume 2*. pp.151-163 (2018). DOI: <https://doi.org/10.1007/978-3-319-67175-8>
- [4] Petfinder.com, a website for pet adoption. <https://www.petfinder.com/>, date accessed 30.06.2021
- [5] Tech4Animals, *DogAge Challenge*, <http://132.75.251.84:3000/~tech4animals/dogchallenge/>, date accessed: 30.06.2021
- [6] Redmon, J., Divvala, S., Girshick, R. & Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition*. (2016). DOI : 10.1109/CVPR.2016.91
- [7] Tan, M. & Le, Q.V. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *Proceedings of the 36th International Conference on Machine Learning, ICML 2019*. pp. 6105-6114 (2019).
- [8] Loshchilov I. & Hutter F. Decoupled Weight Decay Regularization. *International Conference on Learning Representations 2017*. (2017).
- [9] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J. & Houlsby, N. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. (2020). DOI: <https://doi.org/10.48550/arXiv.2010.11929>
- [10] Zamansky, A., Sinitca, A., Kaplun, D., Dutra, L. & Young, R., Automatic Estimation of Dog Age: The DogAge Dataset and Challenge. *Artificial Neural Networks And Machine Learning - ICANN 2019: Image Processing*. pp. 421-426 (2019). DOI: 10.1007/978-3-030-30508-6
- [11] Caputa, J., Łukasik, D., Wielgosz, M., Karwatowski, M., Frączek, R., Russek, P. & Wiatr, K. Fast Pre-Diagnosis of Neoplastic Changes in Cytology Images Using Machine Learning. *Applied Sciences*. **11** (2021). DOI: <https://doi.org/10.3390/app11167181>

4th International Workshop on Artificial Intelligence in Machine Vision and Graphics

THE main objective of the 4th Workshop on Artificial Intelligence in Machine Vision and Graphics (AIMaViG'22) is to provide an interdisciplinary forum for researchers and developers to present and discuss the latest advances of artificial intelligence in the context of machine vision and computer graphics. Recent advancements in artificial intelligence resulted in the rapid growth of both methods and applications of machine learning approaches in computer vision, image processing, and analysis. The development of parallel computing capabilities in the first decade of the 21st century that boosted the development of deep neural networks became a real gamechanger in machine vision. The workshop covers the whole range of AI-based theories, methods, algorithms, technologies, and systems for diversified and heterogeneous areas related to digital images and computer graphics.

TOPICS

The topics and areas include but are not limited to:

- image processing and analysis:
 - image enhancement,
 - linear and non-linear filtering,
 - object detection and segmentation,
 - shape analysis,
 - scene analysis and modeling,
 - scene understanding,
- machine learning for vision and graphics:
 - pattern recognition,
 - deep neural models,
 - convolutional networks,
 - recurrent networks,
 - graph networks,
 - generative adversarial networks,
 - neural style transfer,
 - deep reinforcement learning,
- machine vision:
 - image acquisition,
 - stereo and multispectral imaging,
 - embedded vision,
 - robotic vision,
- image theory:
 - computational geometry,
 - image models and transforms,
 - modeling of human visual perception,
 - visual knowledge representation and reasoning,
- visualization and computer graphics:

- data-driven image synthesis,
- graphical data presentation,
- computer-aided graphic arts and animation,
- applications:
 - innovative uses of graphic and vision systems,
 - image retrieval,
 - autonomous driving systems,
 - remote sensing,
 - digital microscopy,
 - security and surveillance systems,
 - document analysis,
 - OCR systems.

TECHNICAL SESSION CHAIRS

- **Iwanowski, Marcin**, Warsaw University of Technology, Poland
- **Kwaśnicka, Halina**, Wrocław University of Science and Technology, Poland
- **Śluzek, Andrzej**, Khalifa University, United Arab Emirates

PROGRAM COMMITTEE

- **Andrysiak, Tomasz**, Bydgoszcz University of Science and Technology, Poland
- **Angulo, Jesús**, Mines ParisTech, France
- **Cyganek, Bogusław**, AGH University of Science and Technology, Poland
- **Kasprzak, Włodzimierz**, Warsaw University of Technology, Poland
- **Kwolek, Bogdan**, AGH University of Science and Technology, Poland
- **Okarma, Krzysztof**, West Pomeranian University of Technology, Poland
- **Olszewski, Dominik**, Warsaw University of Technology, Poland
- **Palus, Henryk**, Silesian University of Technology, Poland
- **Subbotin, Sergey**, Zaporozhye National Technical University, Ukraine
- **Tomczyk, Arkadiusz**, Lodz University of Technology, Poland

On the Feasible Regions Delimiting Natural Human Postures in a Novel Skeletal Representation

Simon B. Hengeveld* A. Mucherino,*

*IRISA, University of Rennes 1, Rennes, France.

simon.hengeveld@irisa.fr, antonio.mucherino@irisa.fr

Abstract—The de facto standard for storing human motion data on a computer involves a representation based on Euler angles. This representation, while effective, has several shortcomings. Triplets of Euler angles are not unique, and the same posture may be expressed using different combinations of angles. Furthermore, many possible Euler angle triplets correspond to unnatural positions for human joints. This means that, in general, a large part of the representational space remains unused. In this paper, we further investigate a recently proposed representation inspired by molecular representations. It uses only two (instead of three) degrees of freedom per joint: a vector and a torsion angle. Using the two key ingredients of this new representation, we present a complete analysis of the Graphics Lab Motion Capture Database. The data found in this analysis provide us with some powerful insights about natural and unnatural human postures in human motions. These insights can potentially lead to possible constraints on human motions which may be used to more effectively solve open problems in the computer graphics community, most notably the problem of (human) motion adaptation.

I. INTRODUCTION

SEVERAL open problems in computer graphics deal with human motions [2], [3], [9], in which these motions may come from motion capture recordings. In human motions, we have a skeletal character which changes its postures over time. The anatomy of the character in these motions can be defined as a graph $G = (V, E)$, in which the vertex set V represents the *joints* and the edge set E represents the *bones* of the human skeleton. In this work, and generally in the context of human motions, these graphs G are regarded as trees, in which every joint $v \in V$ has a unique *parent* joint, assigned by a function $p : v \in V \setminus \{v_0\} \rightarrow p(v)$. Note that because G is a tree, we have that the number of bones $|E| = |V| - 1$.

In order to complete the representation of our characters, we need the function

$$\chi : v \in V \rightarrow \chi(v) \in \mathbb{R}^3,$$

which assigns a three-dimensional offset from to every joint of the character to its unique parent joint. The real value $\|\chi(v)\|$ corresponds to the length of the bone $\{u, v\} \in E$, where the symbol $\|\cdot\|$ represents the Euclidean norm. This means that this function χ together with the anatomy G lets us define the *morphology* of the skeleton (G, χ) [8]. In graph theory, the pair (G, χ) is generally referred to as “skeletal structure”.

Furthermore, if we add a fictive root joint v_0 to the tree G and fix it at position $(0, 0, 0)$, we can use the offsets between

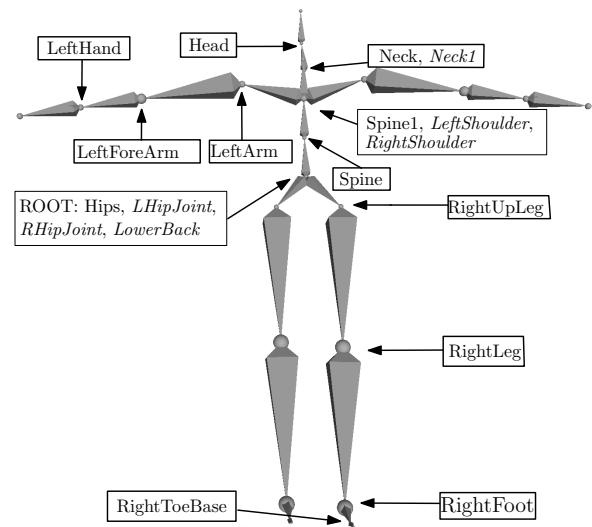


Fig. 1: An example of skeletal structure (G, χ) . This T-pose is the commonly used initial posture of the human motions, particularly in BVH files [6]. These labels we use for the joints originate from files composing the motion database we use in our analysis. Joints with $|\chi| = 0$ are shown next to their parent joint, marked in italic.

a joint v and its parent defined by χ to find a realization of the *initial posture* x_0 of the skeleton. If we let p be the function that pairs the parent to each vertex $v \in V$, we can define the realization of the initial posture as follows:

$$x_0 : v \in V \rightarrow \begin{cases} (0, 0, 0) & \text{if } v = v_0, \\ x_0(p(v)) + \chi(v) & \text{otherwise.} \end{cases}$$

Fig. 1 shows a commonly used default position for the skeletal motions considered in this paper. The motion itself is then defined by the changing positions of the n joints in V over time, where the time is generally defined as a sequence of m frames $t \in T$, with $T = \{1, 2, \dots, m\}$. A possible simple choice for representing the motion of the character would be to use Cartesian coordinates to assign a three-dimensional position to each joint changing over time:

$$x_t : v \in V \rightarrow x_t(v) \in \mathbb{R}^3. \quad (1)$$

However, this Cartesian representation does not explicitly encapsulate the information about the morphology of skeleton.

The standard way of describing human motions utilizes an Euler angle representation ρ . At every frame t , we assign a triplet of Euler angles θ (pitch), ϕ (roll) and η (yaw) to every bone of the skeletal structure representing our character. Together with the known offset between $p(v)$ and v , we can define a transformation matrix $M_t(v)$ that takes into consideration both the translation data from $\chi(v)$ and the given Euler angles. The formula capable to converting the Euler angles in absolute positions for the joints is:

$$\rho_t : v \in V \longrightarrow \begin{cases} (0, 0, 0) & \text{if } v = v_0, \\ \prod_{u \in P(v)} M_t(u) [0, 0, 0, 1]^T & \text{otherwise,} \end{cases}$$

where $P(v)$ is set of vertices u that form the unique path from v_0 to v over the tree structure of the graph. For a more detailed discussion of the transformations involved in the Euler representations (with a particular focus on the standard BVH file format), the reader is referred to [6].

In this work, we consider the vector-torsion angle representation proposed in [4] to represent our human skeletons, and present an in-depth statistical analysis, over a large database, aiming at identifying the feasible regions in the vector-torsion angle space for all joints forming a human character. This is a continuation of the work previously proposed in [4], in which we had only presented some preliminary results for this analysis. To the best of our knowledge, this is the first time an analysis of this kind on the full skeleton has been performed in the relation to human motions and motion capture in general.

The rest of the paper is organized as follows. In Section II, we recall the main definitions regarding our vector-torsion angle representation of the human skeleton. In Section III, we present our analysis on a very-well known motion capture database where the new vector-torsion angle representation is employed. This complete analysis allows us to identify constraints for each of the joints of the skeleton. As finally discussed in Section IV, the results of the presented analysis are likely to have a positive impact on computer graphics applications such as motion adaptation.

II. THE VECTOR-TORSION ANGLE REPRESENTATION

We briefly summarize in this section the main ideas behind the vector-torsion angle representation initially proposed in [4]. Recall that the graph G is a tree representing the anatomy of the human character, and that, together with the offset function χ , it defines a skeletal structure (G, χ) describing the full morphology of the character. By following the natural vertex order given by the structure of G , we can define two angles, a *vector* angle and a *torsion* angle ω_v , for every vertex $v \in V$ which has at least three ancestors.

Definition 1 Given a skeletal structure (G, χ) and one realization x , the vector angle ζ_v for the joint v in this realization is the **smallest** angle (in the range $[0, 180^\circ]$) formed by the line passing through $x((p \circ p)(v))$ and $x(p(v))$, and the line passing through $x(p(v))$ and $x(v)$.

Definition 2 Given a skeletal structure (G, χ) and one realization x , the torsion angle ω_v for the joint v in this realization is the **clockwise** angle (in the range $[0, 360^\circ]$) formed by the plane defined by $x((p \circ p \circ p)(v))$, $x((p \circ p)(v))$ and $x(p(v))$, and the plane defined by $x((p \circ p)(v))$, $x(p(v))$ and $x(v)$.

When a realization x preserves the morphology of the character, we can use this pair of angles combined with the bone lengths (defined by χ) to find the Cartesian coordinates of any joint v that has at least three ancestors. We point out that the idea of using these two angles, while novel in relation to motions, stems from work in the field of Molecular Biology. There, the angles are in fact used in the context of proteins and other molecules in order to differentiate between molecular conformations [1], [10]. The value that the vector-torsion representations has to offer to represent motions can be summarized in the following four advantages: (i) the combination of a vector angle with a torsion angle cannot lead to any representation singularities, (ii) there exists a bijective correspondence between the value of the angles and the positions in space for the joints, (iii) it exhibits only two degrees of freedom for recovering the same joints positions of the skeletal representations, as the triplets of Euler angles are capable to do with three degrees of freedom, and (iv) it allows us to empirically constrain the feasible (and mostly continuous!) regions in the vector-torsion angle space where only natural postures for the human skeleton can be found.

For more details about the vector-torsion representation, the reader is referred to our original publication [4].

III. MOTION ANALYSIS

In this section we present an analysis of the Graphics Lab Motion Capture Database¹, a large database of human motions resulting from recordings using motion capture. The database contains 2436 motion files which sum to a total of more than four million frames. In the following, we will refer to human joints with labels such as Hips, RightShoulder, LeftLeg and others, which we take from the data file forming the motion database.

Using the vector-torsion representation briefly presented in Section II, we conducted a statistical analysis of the vector angles ζ_v^t and torsion angles ω_v^t of every applicable joint v at every frame t of these four million frames. Using the resulting data from this experiment, we generated a heat-map scatter-plot for each joint v .

Even though it is not possible to define the vector and torsion angles for the joints having fewer than three ancestors, we also performed the analysis on these joints. In fact, there is no need in our analysis to build up human postures, but only to *look at* the available postures of the database. Therefore, for those joints missing a sufficient number of ancestors, we have simply defined a different set of “reference joints” in the graph G , that we subsequently use for defining the vector and torsion angles. Table I shows the complete set of joints for which we performed our analysis, together with the list of

¹<https://mocap.cs.cmu.edu>

TABLE I: The three reference joints for each joint involved in the analysis. We suppose that the reference joint with smallest numerical label is the closest; the one with largest numerical label is instead the farthest. In some cases, the reference joints are not the joint ancestors implied by the graph structure.

<i>joint</i>	<i>ref#1</i>	<i>ref#2</i>	<i>ref#3</i>
Head	Neck	Spine1	Spine
Neck	Spine1	Spine	Hips
Spine1	Spine	Hips	RightUpLeg
Spine	Hips	RightUpLeg	RightLeg
LeftArm	Spine1	Spine	Hips
RightArm	Spine1	Spine	Hips
LeftForeArm	LeftArm	Spine1	Spine
RightForeArm	RightArm	Spine1	Spine
LeftHand	LeftForeArm	LeftArm	Spine1
RightHand	RightForeArm	RightArm	Spine1
LeftUpLeg	Hips	Spine	Spine1
RightUpLeg	Hips	Spine	Spine1
LeftLeg	LeftUpLeg	Hips	Spine
RightLeg	RightUpLeg	Hips	Spine
LeftFoot	LeftLeg	LeftUpLeg	Hips
RightFoot	RightLeg	RightUpLeg	Hips
LeftToeBase	LeftFoot	LeftLeg	LeftUpLeg
RightToeBase	RightFoot	RightLeg	RightUpLeg

three reference joints used for computing the vector and the torsion angles.

Other joints are however excluded from our analysis. These are all the joints which share the same global position with some other joints, because their offset to the parent has zero length. Although this may sound like a contradiction, these joints actually have the purpose of modifying the orientation (the corresponding Euler angles are non-zero) of the entire set of subsequent bones on the current skeleton branch. Therefore, joints with $|\chi_v| = 0$ are omitted, and they are not counted as ancestors of other joints either.

We present several scatter-plots obtained in the analysis described above, starting at the head of the skeleton, working our way down to the feet. The following figures show a total of 18 plots for different joints of our human skeleton, with the vector angles on the x -axis and the torsion angles on the y -axis. In these plots, points tending to the warmer colors correspond to pairs of angles that were found more frequently.

Fig. 2 collects the first set of 12 scatter-plots. The name of joints related to the presented plots are given in the plot itself. The first joint we considered, the Head joint, is only able to perform limited movements in the space defined by the vector and torsion angle, as expected. Roughly speaking, only one third of this space is actually feasible for this joint. Moreover, the warmer part of the scatter-plot indicates that the most common posture for this joint is when the two angles are close to 180° , which is compatible with an erected posture for the upper body part.

While the Neck joint exhibits a pattern very similar to the one of the Head joint, we notice that the two joints involved in the modeling of the human spine (Spine and Spine1 joints) admit an even smaller feasible space. This is particularly true for the Spine1 joint (see Fig. 1 to identify the exact location of the joint), where a large part of the scatter-plot remained

“immaculate white”, which is, the combinations of vector and torsion angles in those white areas are completely infeasible for a human spine.

The LeftArm and RightArm joints exhibit a quite constrained pattern as well, which is similar to those found for some of the previous joints but shifted in the center of the vector angle axis (the x -axis). This corresponds to saying that the angle formed by the spine and the one of these two joints is in most of the cases close to 90° . Notice in fact that these two joints share the same global Cartesian positions with the LeftShoulder and RightShoulder joints. We can also remark that the two scatter-plots are symmetric w.r.t. the axis parallel to the x -axis and passing through the torsion angle value 180° .

The expected flexibility for the human arms is reflected in the two corresponding scatter-plots, the ones related to the LeftForeArm and RightForeArm joints. This is in fact the first pair of joints that we comment for which the “colored areas” are able to cover more than 50% of the two-dimensional space. Yet, there are still particular combinations of vector and torsion angles that correspond to unnatural postures.

Similarly, the scatter-plots related to the LeftHand and RightHand joints show a quite large range of movement possibilities. This was expected as well. Moreover, the little populated blue areas in these two scatter-plots seem to suggest the extremely high flexibility of the human hand: even if sometimes very uncommon (a few frames of the database may contain them), there exist very special (and still natural) postures that the human hand can take. Therefore, if we take into consideration in full these low-populated areas, we can state that the scatter-plots related to the human hands are the ones that almost cover the entire two-dimensional space.

When stepping down over the joints forming the human legs, we can observe similar patterns. For the LeftUpLeg and the RightUpLeg joints, we can notice that the patterns are similar to those observed for the two upper arm bones. The same applies for the LeftLeg and RightLeg joints (see Fig. 3, even if they seem to admit little larger movement possibilities w.r.t. the corresponding arm joints). The human feet (LeftFoot and RightFoot joints) also exhibit quite large movement possibilities, similarly to what we have observed for the hand joints, but the low-populated areas in the scatter-plots for the feet are much more sparse. This may be consequence of the fact that the human feet lost, during evolution, part of their movement possibilities, but the similarity to the hands seems to be still visible in our figures.

Finally, the scatter-plots of the LeftToeBase and RightToeBase joints show that they are the only joints that can actually span the entire two-dimensional space, but the region is mostly not continuous and most of the vector-torsion combinations are actually placed around the center of the plot, where $\zeta_v = 90^\circ$ and $\omega_v = 180^\circ$.

To sum up, our analysis shows that there exist large differences in flexibility between different joints, and the feasible regions tend to vary a lot on the basis of the nature of each joint. In general, the vector angle seems to be the most restrictive factor. In fact, for joints like the Head, Neck and

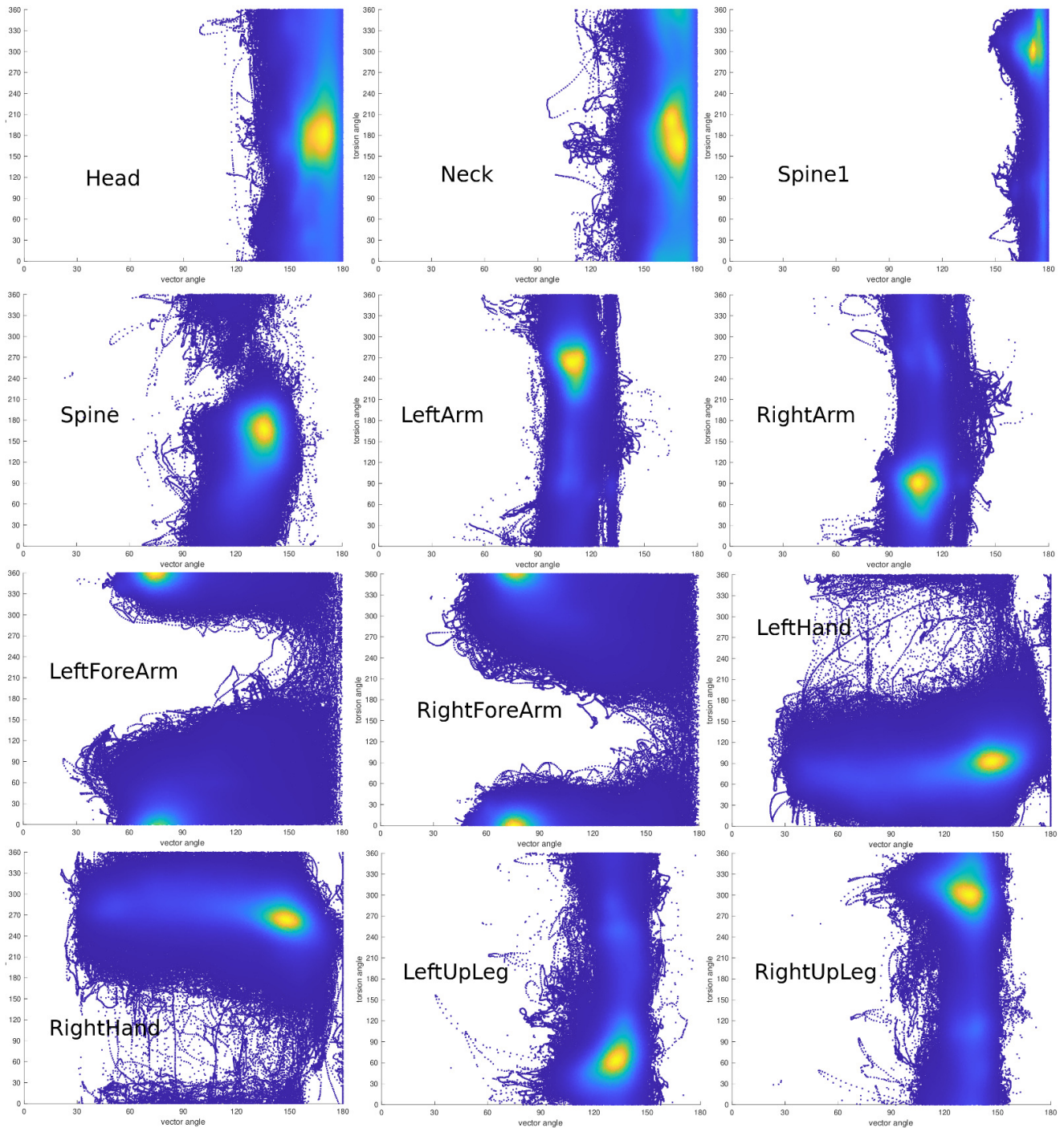


Fig. 2: The first 12 scatter-plots obtained in our analysis.

Spine joints, we see that the feasible vector angle regions are around the 180° mark and do not vary much. The plots for these joints are quite similar, and this makes sense when we look at their respective ancestor joints in the skeleton (see Fig. 1)

Joints on the right and left side of the human body appear

to have very comparable regions, except for the fact that the values of the torsion angles are generally inverted. This is a result from the fact that we use a clockwise rotation to compute the torsion angles between the two planes defined by the quadruplets of joints. This gives rise to the symmetry property mentioned above.

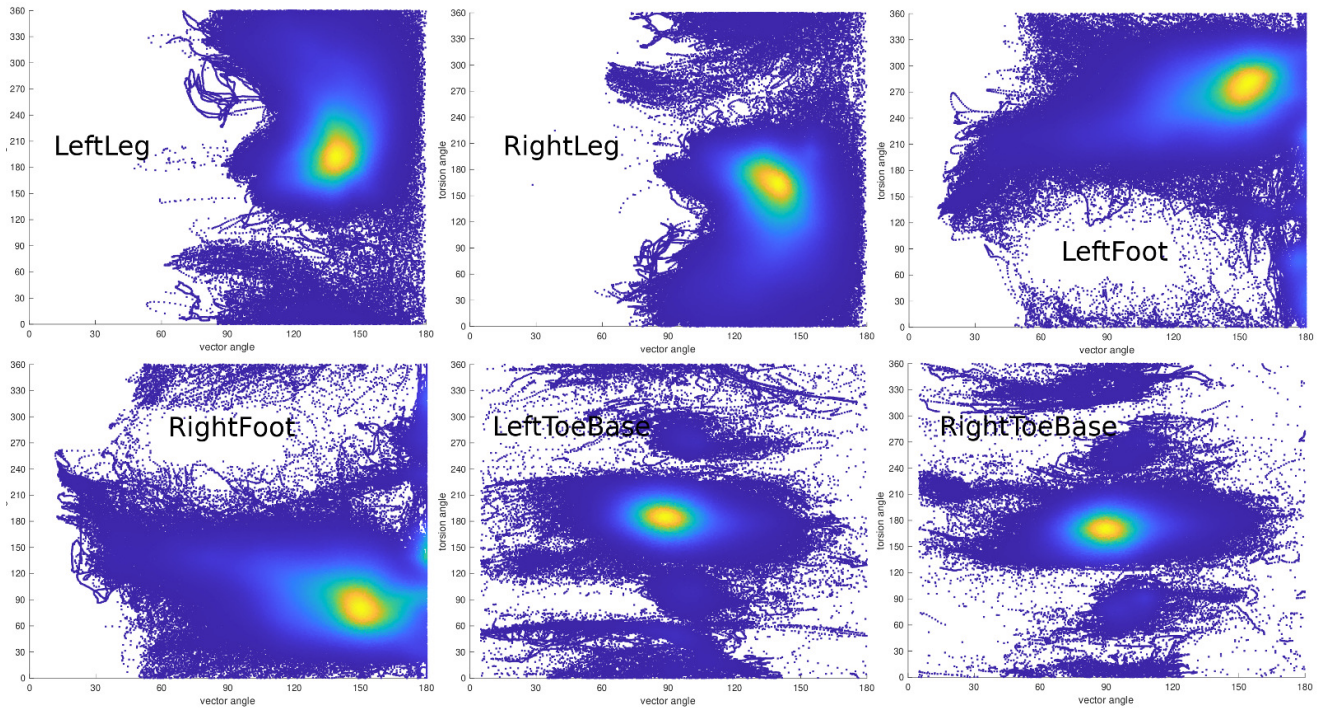


Fig. 3: The remaining 6 scatter-plots obtained in our analysis.

IV. CONCLUSIONS

We have further expanded on the recently proposed vector-torsion angle representation for human motions. Furthermore, we presented an extensive analysis using this representation, in order to find the feasible regions which delimit natural human postures during human motions.

Constraints that can be derived from this analysis are likely to play a very important rule in works on motion adaptation [2], [3]. They may allow us to avoid defining many unnatural human positions in an attempt to create motions satisfying some new constraints, related for example to a change of morphology for the character. When using our vector-torsion representation, the constraints on the values of the vector and torsion angles can directly be imposed; those related to joints with too few ancestors may instead be used for verification. Applying such constraints for motion adaptation, in the context of *dynamical distance geometry* [7], [8], [9], is one of the main directions for future work.

Acknowledgments

This work is partially supported by the international project MULTIBIOSTRUCT funded by the ANR French funding agency (ANR-19-CE45-0019).

REFERENCES

- [1] H. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. Bhat, H. Weissig, I. Shindyalov, P. Bourne, *The Protein Data Bank*, *Nucleic Acids Research* **28**, 235–242, 2000.
- [2] M. Gleicher, *Retargetting Motion to New Characters*. ACM Proceedings of the 25th annual conference on Computer Graphics and Interactive Techniques, 33–42, 1998.
- [3] S. Guo, R. Southern, J. Chang, D. Greer, J.J. Zhang, *Adaptive Motion Synthesis for Virtual Characters: a Survey*, *The Visual Computer* **31**(5), 497–512, 2015.
- [4] S.B. Hengeveld, A. Mucherino, *On the Representation of Human Motions and Distance-based Retargeting*, IEEE Conference Proceedings, Federated Conference on Computer Science and Information Systems (FedCSIS21), Workshop on Computational Optimization (WCO21), Sofia, Bulgaria, 181–189, 2021.
- [5] W. Maurel, D. Thalmann, *Human Shoulder Modeling Including Scapulo-Thoracic Constraint and Joint Sinus Cones*, *Computers & Graphics* **24**, 203–218, 2000.
- [6] M. Meredith, S. Maddock, *Motion Capture File Formats Explained*, Technical Report 211, Department of Computer Science, University of Sheffield, 36 pages, 2001.
- [7] A. Mucherino, D.S. Gonçalves, *An Approach to Dynamical Distance Geometry*, Lecture Notes in Computer Science **10589**, F. Nielsen, F. Barbaresco (Eds.), Proceedings of Geometric Science of Information (GSI17), Paris, France, 821–829, 2017.
- [8] A. Mucherino, D.S. Gonçalves, A. Bernardin, L. Hoyet, F. Multon, *A Distance-Based Approach for Human Posture Simulations*, IEEE Conference Proceedings, Federated Conference on Computer Science and Information Systems (FedCSIS17), Workshop on Computational Optimization (WCO17), Prague, Czech Republic, 441–444, 2017.
- [9] A. Mucherino, J. Omer, L. Hoyet, P. Robuffo Giordano, F. Multon, *An Application-based Characterization of Dynamical Distance Geometry Problems*, *Optimization Letters* **14**(2), 493–507, 2020.
- [10] G.N. Ramachandran, C. Ramakrishnan, V. Sasisekharan, *Stereochemistry of Polypeptide Chain Configurations*, *Journal of Molecular Biology* **7**, 95–104, 1963.
- [11] G.G. Slabaugh, *Computing Euler Angles from a Rotation Matrix*, Technical Report, City University London, 8 pages, 1999.

A lightweight approach to two-person interaction classification in sparse image sequences

Włodzimierz Kasprzak, Paweł Piwowarski
 Warsaw University of Technology
 Institute of Control and Computation Eng.
 ul. Nowowiejska 15/19
 00-665 Warszawa, Poland

Email: {wlodzimierz.kasprzak, pawel.piwowarski.dokt}@pw.edu.pl

Van-Khanh Do
no affiliation

Email: khandovanit@gmail.com

Abstract—A lightweight neural network-based approach to two-person interaction classification in image sequences, based on human skeletons detected in sparse video frames, is proposed. The idea is to use an ensemble of pose classifiers (“experts”), where every expert is trained on different time-indexed snapshots of an interaction. Thus, the expertise of “weak” classifiers is distributed over the time duration of an interaction. The overall classification result is a weighted combination of all the pose experts. Important element of proposed solution is the refinement of skeleton data, based on a merging-of-joints procedure. This allows the generation of reliable features being passed to the artificial neural network. This is the key to our lightweight solution, as ANN resources, needed for feature space transformation, can be significantly limited. Our network model was trained and tested on the interaction subset of the well-known NTU RGB+D dataset, although only 2D skeleton information is used, typical in video analysis. The test results show comparable performance of our method with some of the best so far reported STM- and CNN-based classifiers for this dataset, when they process sparse frame sequences, like we did. The recently proposed multi-stream Graph CNNs have shown superior results but only when processing dense frame sequences. Considering the dominating processing time and resources needed for skeleton estimation in every frame of the sequence, the key to real-time interaction recognition is to limit the number of processed frames.

I. INTRODUCTION

The aim of our work is the analysis of human interactions in specific time-related image sequences. The data can originate from decomposition of video clips onto frames or directly from snapshots of videos posted as image galleries in the Internet. Their common property is the sparsity of time-relevant information (Figure 1).

The approaches to vision-based human activity recognition can be divided into two main categories: activity recognition directly in video data [1] or skeleton-based methods [2], where the 2D or 3D human skeletons are detected first, even by specialized devices, like the Microsoft Kinect.

In early solutions, hand-designed features like edges, contours, Scale-Invariant Feature Transform (SIFT), Histogram of Oriented Gradients (HOG) have usually been used for

detection and localization of human body parts or key points in the image [3], [4].

More recently, Neural Network-based solutions were successfully proposed for solving human pose- and human activity recognition problems, e.g., solutions are based on Deep Neural Networks (DNN) [5], especially on Long-Short Term Memory (LSTM) models and Convolutional Neural Networks (CNN) [6], and more recently on Graph CNNs [7]. CNNs have the capability automatically to learn rich semantic and discriminative features from images and multi-dimensional signals. Furthermore, CNNs can learn both spatial and temporal information from signals and model scale-invariant features as well. Graph CNNs allow efficient implementations of convolution layers when structured data (i.e., graphs) are processed. Some popular solutions to human skeleton estimation (i.e., the detection and localization) in images, based on DNN and CNN models, can be mentioned: OpenPose [8], DeepPose [9] and DeeperCut [10].

Hence, nowadays human action- and interaction recognition in video is most often based on skeleton data extracted from video frames. The state-of-the-art solutions to human action encoding and classification, which process human skeleton data, typically use “heavy” deep neural networks, like 3D CNNs and LSTMs or slightly lightweight Graph CNNs [11], [12].

In this work, we focus on two-person interaction recognition in sparse frame sequences, assuming the existence of skeleton data for key video frames. We took the straightforward idea of extending two-person pose classification of still images to two-person interaction classification in image sequences, by applying an ensemble of pose classifiers [13]. Typically for a classifier ensemble, individual classifiers are “experts” in different parts of the input data domain and the extra weighting network differentiates between subdomains. In our approach, the pose-classifiers are experts at different time stages, while their input space itself (i.e., the spatial image information) is not affecting the fusion weights. By performing a simple time decomposition, we are going to distinguish four subsequent time periods of an interaction process, e.g. start, before midterm, after midterm and final. The final fusion will take the form of a weighted sum of class likelihoods of all the pose classifiers.

This work was supported by “Narodowe Centrum Badań i Rozwoju”, Warszawa, Poland, grant No. CYBERSECIDENT/455132/III/NCBR/2020 - the APAKT project

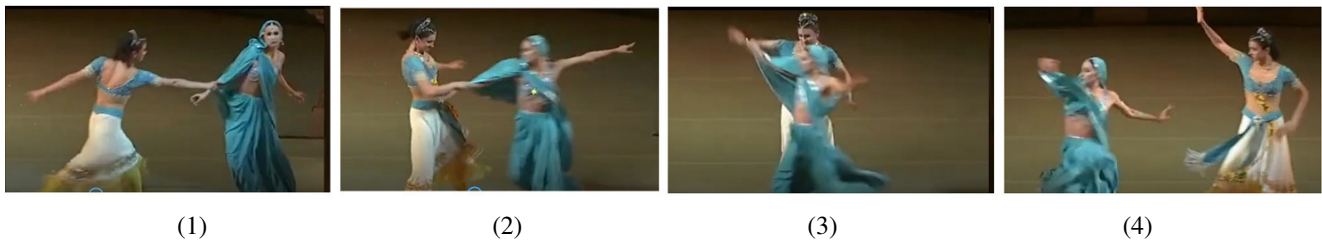


Fig. 1. Example of a sparse sequence of frames from a two-person interaction video

There are 4 remaining sections of this work. Section II refers some recent approaches in human pose, -action and -interaction recognition. Our solution is presented in section III. In section IV, experiments are described, to verify the approach. The classifiers are trained and tested on two datasets: an own human pose image dataset, called "humiact5", and the well-known video dataset for action and interaction, NTU RGB+D [14]. Finally, in section V, we summarize our work and contribution to the subject.

II. RELATED WORK

The recognition of human activities in video is a hot research topic in the last 15 years. Typically, human activity recognition in images and video requires first a detection of human body parts or key-points of a human skeleton. The skeleton-based methods compensate some of the drawbacks of vision-based methods, such as assuring the privacy of persons and reducing the scene lightness sensitivity.

The vast majority of research is based on the use of artificial neural networks. However, more classical approaches have also been tried, such as the SVM (e.g. [15], [16]). Yan et al. [17] used multiple features, like a "bag of interest points" and a "histogram of interest point locations", to represent human actions. They proposed a combination of classifiers in which AdaBoost and sparse representation (SR) are used as basic algorithms. In the work of Vemulapalli et al. [18] human actions are modeled as curves in a Lie group of Euclidean distances. The classification process is using a combination of dynamic time warping, Fourier temporal pyramid representation and linear SVM.

Thanks to higher quality results, artificial neural networks are replacing other methods. Thus, the most recently conducted research in the area of human activity classification differs only by the proposed network architecture. Networks based on the LSTM architecture or a modification of this architecture (a ST-LSTM network with trust gates) were proposed by Liu et al. [19] and Shahroudy et al. [14]. They introduced so called "Trust Gates" for controlling the content of an LSTM cell and designed an LSTM network capable of capturing spatial and temporal dependencies at the same time (denoted as ST-LSTM). The task performed by the gates is to assess the reliability of the obtained joint positions based on the temporal and spatial context. This context is based on the position of the examined junction in the previous moment (temporal context) and the position of the previously studied junction in

the present moment (spatial context). This behavior is intended to help network memory cells assess which locations should not be remembered and which ones should be kept in memory. The authors also drew attention to the importance of capturing default spatial dependencies already in the skeleton data. They have experimented with different mappings of the a joint's set to a sequence. Among the, they mapped the skeleton data into a tree representation, duplicating joints when necessary to keep spatial neighborhood relation, and performed a tree traversal to get a sequence of joints. Such an enhancement of the input data allowed an increase of the classification accuracy by several percent.

The work [20] introduced the idea of applying convolutional filters to pseudo-images in the context of action classification. A pseudo-image is a map (a 2D matrix) of feature vectors from successive time points, aligned along the time axis. Thanks to these two dimensions, the convolutional filters find local relationships of a combined time-space nature. Liang et al. [21] extended this idea to a multi-stream network with three stages. They use 3 types of features, extracted from the skeleton data: positions of joints, motions of joints and orientations of line segments between joints. Every feature type is processed independently in an own stream but after every stage the results are exchanged between streams.

Graph convolutional networks are currently considered as a natural approach to the action (and interaction) recognition problem. They are able to achieve high quality results with only modest requirements of computational resources. "Spatial Temporal Graph Convolutional Networks" [22] and "Actional-Structural Graph Convolutional Networks" [23] are examples of such an solution.

Another recent development is the pre-processing of the skeleton data in order to extract different type of information (e.g., information on joints and bones, and their relations in space and time). Such data streams are first separately processed by so called multi-stream neural networks and later fused to a final result. Examples of such solutions are the "Two-Stream Adaptive Graph Convolutional Network" (2S-AGCN) and the "Multistream Adaptive Graph Convolutional Network" (AAGCN), proposed by Shi et al. [24], [25].

One of the best performances on the NTU RGB+D interaction dataset is reported in the work of Perez et al. [26]. Its main contribution is a powerful two-stream network with three-stages, called "Interaction Relational Network" (IRN).

The network input are basic relations between joints of two interacting persons tracked over the length of image sequence. An important step is the initial extraction of relations between pairs of joints - both distances between joints and their motion are obtained. The neural network makes further encoding and decoding of these relations and a final classification. The first stream means the processing of within-a-person relations, while the second one - between-person relations. The use of a final LSTM with 256 units is a high-quality version of the IRN network, called IRN-LSTM. It allows to reason over the interactions during the whole video sequence - even all frames of the video clip are expected to be processed. In the basic IRN, a simple densely-connected classifier is used instead of the LSTM and a sparse sequence of frames is processed.

The currently best results are reported by Zhu et al. [27], where two new modules are proposed for a baseline 2S-AGCN network. The first module extends the idea of modelling relational links between two skeletons by a spatio-temporal graph to a "Relational Adjacency Matrix (RAM)". The second novelty is a processing module, called "Dyadic Relational Graph Convolution Block", which combines the RAM with spatial graph convolution and temporal convolution to generate new spatial-temporal features.

From the analysis of the recent most successful solutions, we can draw three main conclusions:

- 1) using an analytic preprocessing of skeleton-data to extract meaningful information and cancel noisy data, either by employing classic functions or learnable function approximations (e.g. relational networks);
- 2) preferring light-weight solutions by employing background (problem-specific) knowledge, i.e. using graph CNNs instead of CNN, CNNs with 2-D kernels instead of 3-D CNN;
- 3) a video clip containing a specific human action or interaction can be processed alternatively as a sparse or dense frame sequence, where sparse sequence is chosen to achieve real-time processing under limited computational resources, while the processing of a dense sequence leads to better performance.

III. THE APPROACH

A. Structure

The input data for our interaction classifier is a sequence of sparse video frames. Assuming, a video clip is given the start and end of an interaction should be detected first. Then, the video clip is split into some number M of consecutive time intervals (e.g. $M = 16$). From each interval one frame is selected for classification. Assume, that $M = N \cdot m$, where N is a period of time, while m the number of frames in one period. We may distinguish $N = 4$ periods: start, 1-st intermediate, 2-nd intermediate and final. To the classification of frames from a single period, a separate pose classifier (the "expert") is dedicated. As shown in Figure 2, the proposed solution consists of several processing stages:

- 1) *Skeleton estimation*: the OpenPose net [28] is applied to detect human skeletons with their 2D joints in an RGB image (a video frame);
- 2) *Feature engineering*: a *keypoint enhancement algorithm* is proposed in order to get more reliable two sets of skeleton joints from the OpenPose results; next, *feature vectors* are extracted from the refined joints.
- 3) *Pose classifier training*: several lightweight, densely-connected MLP networks are trained - every one is a "weak" classifier.
- 4) *Model evaluation*: alternative network models are evaluated, to find the optimal model configuration and training parameter. A *Keras-tuner* [29] - the *RandomSearch* algorithm [30] is applied to find optimal hyper-parameter settings.
- 5) *Ensemble classifier*: a dense gain network is also trained to learn the weights for results of individual pose classifiers. Two versions of the final classifier are implemented - one with fixed weights and one with learned weights.
- 6) *Model testing*: after accumulating the pose class likelihoods over the frame sequence the final most likely interaction class is selected as the winner. Two datasets - an own *humiact5* and the RGB subset of the NTU RGB+D dataset, are used to evaluate the created models.

B. Skeleton estimation

In the paper [8], a multi-person 2D pose estimation architecture was proposed based on part affinity fields (PAFs). The work introduced an explicit nonparametric representation of the keypoint association which encodes both position and orientation of the human limbs. The designed architecture can learn both human keypoint detection and association using heatmaps of human key-points and part affinity fields respectively. It iteratively predicts part affinity fields and part detection confidence maps. The part affinity fields encode part-to-part association including part locations and orientations. In the iterative architecture, both PAFs and confidence maps will be iteratively refined over successive stages with intermediate supervision at each stage. Subsequently, a greedy parsing algorithm is employed to effectively parse human poses. The work ended up releasing the OpenPose library, the first real-time system for multi-person 2D pose estimation [28]. In our research, we use the core block of OpenPose, the "body_25 model", to extract 25 human key-points in images. The result is an 25-elementary array, providing 2D image coordinates and confidence score for every keypoint.

C. Feature engineering

From the (eventually more than two) sets of skeleton joints, detected in the image by OpenPose, the two main actors are selected based on size measure. A total variability of skeleton keypoint locations is calculated for every skeleton and the two with the highest variability are chosen for feature engineering.

1) *Skeleton enhancement*: There are cases where OpenPose wrongly splits one human region into different regions due to

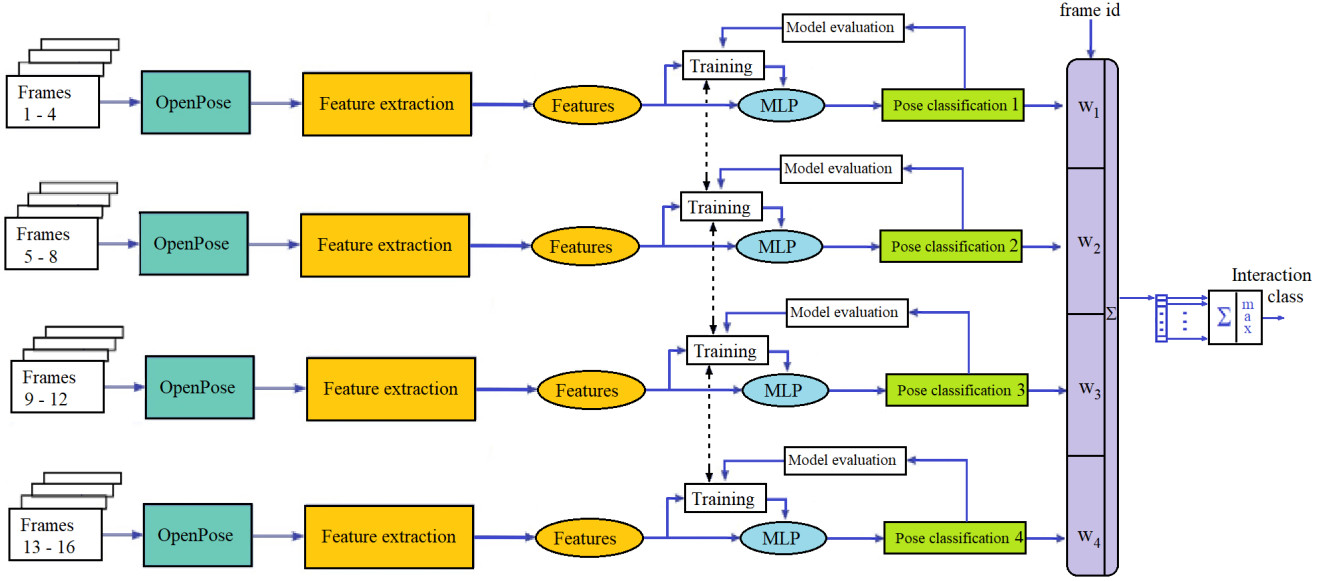


Fig. 2. General structure of our approach

occlusion, low resolution, or complex visual context. Therefore, we developed a keypoint (i.e., skeleton joints) merging and replacement algorithm. In the first step, we try to merge sets of joints, where applicable, to produce finer skeleton joints (see Figure 3).

Two calculations are made for each pair of sets including the number of intersection points of the two sets and the distance between them. The intersection indicator value is scaled by the number of points of the smaller set. The distance calculation takes their two mean points and two standard deviation values into account. These calculated values then will be compared with corresponding thresholds to decide whether the two sets are going to be merged or not. In case merging conditions are met, the intersection points of the two sets will be treated in the following way: the data points with higher probability will be kept and the lower ones will be ignored.

For the sake of clarity, Figure 4 illustrates the merging procedure of two specific sets, A and B , based on the assumption that they come from the same person in the image. The bigger set A is missing key-points for the left leg, while the smaller set B includes these key-points. The mean points (center of gravity) of A and B are m_A and m_B , respectively, the standard deviations of joints locations for A and B are $[std_{A,x}, std_{A,y}]$ and $[std_{B,x}, std_{B,y}]$, respectively.

The conditions for a merging action are as follows:

$$\frac{|A \cap B|}{|B|} \leq \theta_1 \quad (1)$$

$$\frac{|m_{A,x} - m_{B,x}|}{std_{A,x} + std_{B,x}} + \frac{|m_{A,y} - m_{B,y}|}{std_{A,y} + std_{B,y}} \leq \theta_2 \quad (2)$$

where θ_1, θ_2 are intersection threshold and distance threshold, respectively.

After a merging action has been performed, the remaining joints in the smaller set (call it S) can eventually replace low-confident, corresponding joints in a subset B_s of the big set B . To decide about this, the following values are considered: the normalized Euclidean distance between the smaller set joints and the corresponding candidate joints of the subset B_s , the average confidence of all candidate joints in the small set S and the average confidence of the corresponding joints in the bigger set (Figure 5).

Let N -elementary sets S and B_s of corresponding joints are given, considered for possible replacement. Standard deviation coefficients of the smaller set joints locations are $[std_{S,x}, std_{S,y}]$. Let the confidence value of a joint j be denoted as $P(j)$. The conditions for a joints replacement are as follows:

$$\frac{1}{std_{S,x} + std_{S,y}} \sum_{i=1}^N \sqrt{(x_{S_i} - x_{B_{s_i}})^2 + (y_{S_i} - y_{B_{s_i}})^2} \leq \theta_3 \quad (3)$$

$$\frac{1}{N} \sum_{i=1}^N P(S_i) \geq \theta_4 \quad (4)$$

$$\frac{1}{N} \sum_{i=1}^N P(B_{s_i}) \leq \theta_5 \quad (5)$$

where θ_3 is the normalized Euclidean distance threshold, θ_4 - the confidence threshold for S and θ_5 - the confidence threshold for B_s .

The skeletons, which remain after the merging and replacement steps, will be ordered by their bounding box size in descending order. With (w, h) representing width and height of a bounding box, the $score = w \cdot h$. The two sets with highest score will be kept and used further in the feature extraction step.

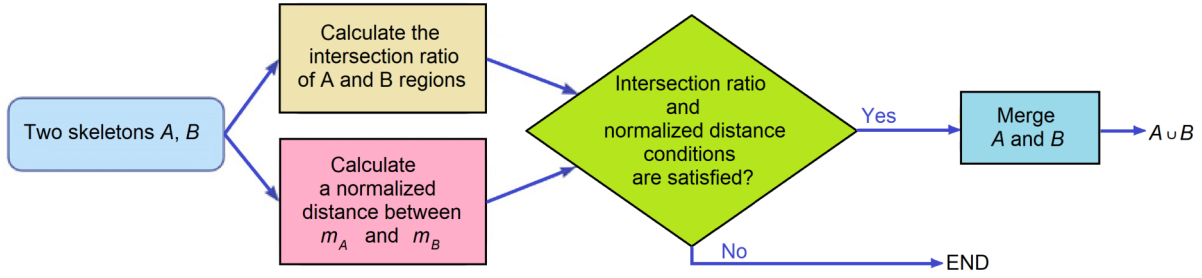


Fig. 3. The skeleton merging step.

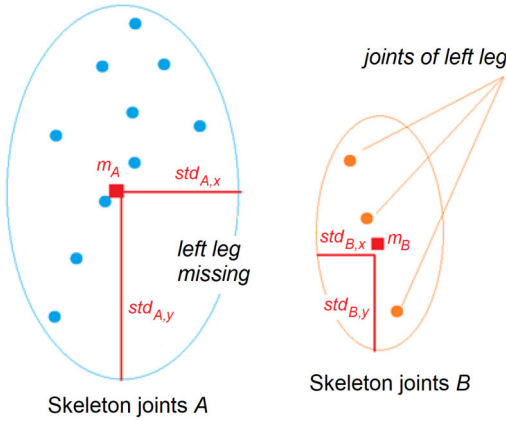


Fig. 4. Illustration of a skeleton merging situation.

2) *Feature extraction*: Feature extraction means the calculation of normalized distances between pairs of joints from two skeletons, tracked in the frame sequence of a video clip. First, the distance between two middle-of-spine points of two human skeletons is calculated and normalized by the length of the spine I_1 of the first person, giving the distance feature (Figure 6). Then, every set of joints is independently normalized by: translating the local coordinate system to the middle-of-spine point O_1 or O_2 , rotating the points so that the spine segment (connecting joint 1 with joint 8) is parallel to the Y axis of local system, and finally, scaling the point coordinates by the spine length I_i .

Denote by $\mathbf{H}_1, \mathbf{H}_2$ the skeletons of the first and the second human; O_1, O_2 - the centers of spine segments of the first and second human, respectively; l_1 - the length of the spine segment of the first human; α_1, α_2 - the rotation angles to make corresponding spine segments parallel to the Y axes of local Cartesian coordinate systems. The *distance* feature is calculated as the distance between local system origins, O_1, O_2 , normalized by the length l_1 :

$$d = \frac{\text{distance}}{l_1} \quad (6)$$

The normalization of joints coordinates (translation to local system, rotation, scaling) is performed independently for every set $\mathbf{H}_1, \mathbf{H}_2$. Let $\mathbf{p}_i = (p_{i,x}, p_{i,y})$ denotes the image

coordinates of a joint from skeleton $\mathbf{H}_i, (i = 1, 2)$. The normalization of this joint is given as follows:

$$\mathbf{p}'_i = (p'_{i,x}, p'_{i,y}) = (p_{i,x} - O_{i,x}; p_{i,y} - O_{i,y}), \quad i = 1, 2 \quad (7)$$

$$\begin{pmatrix} p''_{i,x} \\ p''_{i,y} \end{pmatrix} = \begin{bmatrix} \cos(\alpha_i) & -\sin(\alpha_i) \\ \sin(\alpha_i) & \cos(\alpha_i) \end{bmatrix} \begin{pmatrix} p'_{i,x} \\ p'_{i,y} \end{pmatrix}, \quad i = 1, 2 \quad (8)$$

$$(p'''_{i,x}, p'''_{i,y}) = \left(\frac{p''_{i,x}}{w_i}, \frac{p''_{i,y}}{h_i} \right), \quad i = 1, 2 \quad (9)$$

3) *Feature vector*: Both the OpenPose (applied for our RGB dataset) and the built-in skeleton detector from Kinect v2 (generating the skeleton data in the NTU RGB+D dataset) deliver person skeletons of 25 joints. By analysing a small skeleton data subset, we found that the data for joints numbered from 15 to 24, corresponding to "small" parts, like fingers, are very often missing. Thus, we use only joints numbered from 0 to 14. The feature vector obtained from skeleton data of a single frame has 61 dimensions as there are $15 \text{ joints} \times 2 \text{ coordinates} \times 2 \text{ sets}$ and one distance feature. Assuming that we have selected m frames for analysis, we get a map of $m \times 61$ features.

D. Pose classifier training and evaluation

The feature data is fed to several MLP-based pose classifiers. We use fully-connected MLP architecture with variants of several hyper-parameters: the number of hidden layers of the network can vary from 1 to 3, different activation functions (ReLU and/or sigmoid) may be chosen, as well as the number of neurons in hidden layers and the learning rate can vary. The ANN is implemented using Keras [29].

Automated hyper-parameter tuning [31] is a crucial step during ANN model training to increase the model's performance. We perform a hyper-parameter search during training using the *Random Search* algorithm, offered in Keras [30]. For both datasets the hyper-parameter search space is defined as:

$$S_{\text{search}} = [a_{\text{fun}}, l_{\text{rate}}, n_{\text{layer}}, n_{\text{neur}}], \quad (10)$$

where the entries are: activation function, $a_{\text{fun}} \in \{\text{relu}, \text{sigmoid}\}$, learning rate, $l_{\text{rate}} \in [10^{-5}, 10^{-2}]$, number of hidden layers, $n_{\text{layer}} \in \{1, 2, 3\}$, number of neurons in hidden layer, $n_{\text{neur}} \in \{100, 200, \dots, 1000\}$.

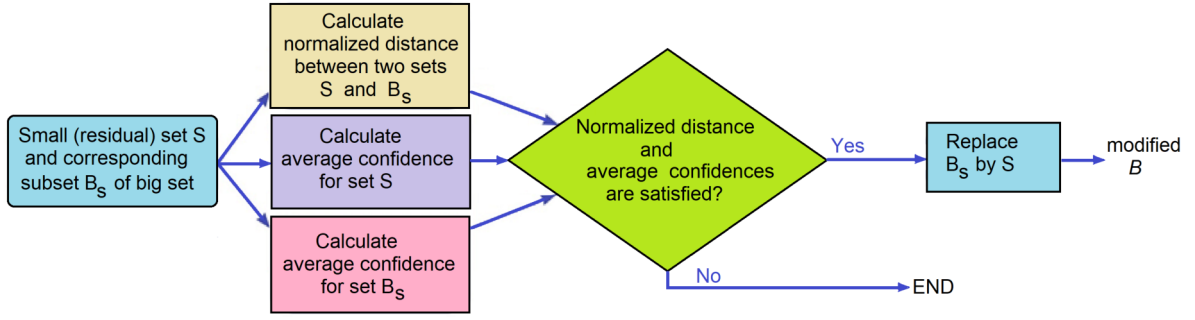


Fig. 5. The joints replacement step

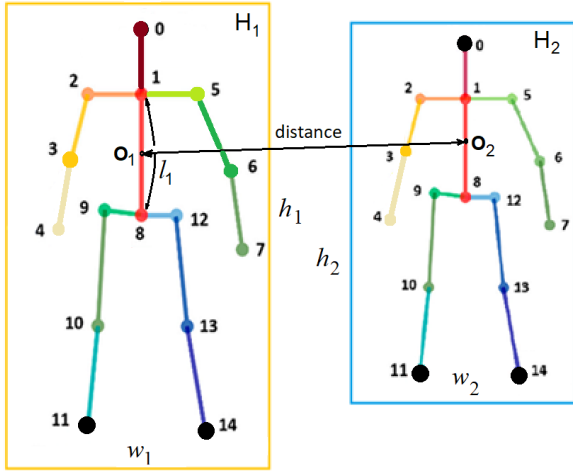


Fig. 6. The normalization elements for two sets of joints

E. Ensemble classifier

As mentioned earlier, every pose classifier is an "expert" to recognize snapshots taken during different time period of an interaction. In practice, the training of such an assembly is performed at the same time, but 3 out of 4 "expert" networks are always in a dropout mode. The actually updated network depends on the time period the current input frame belongs to.

In the testing process, the interaction class is known after the entire frame sequence - from a single video clip - has been classified and the results of individual pose classifiers were accumulated. The likelihood of every interaction class comes from an aggregation of pose class likelihoods, as a weighted sum of pose likelihoods, for frames indexed from $t=0$ to $t=T$.

1) *Fixed gains*: In a hand-crafted form we define the aggregation of likelihoods, obtained by particular pose classifiers ($i = 1, 2, 3, 4$) for frames ($t = 1, 2, \dots, N$), the $\Pr_{pose_i}(t) - s$, as follows:

$$\begin{aligned} \mathbf{S} = \sum_{t=0}^T & [\Pr_{pose_1}(t) \cdot \max(0, (T/2 - t)/T) + \\ & + \Pr_{pose_2}(t) \cdot \min(0.5, t/T) + \\ & + \Pr_{pose_3}(t) \cdot \min(0.5, (T - t)/T) + \\ & + \Pr_{pose_4}(t) \cdot \min(0, (t - T/2)/T)] \end{aligned} \quad (11)$$

2) *The gain network*: In the trained case, the gain network provides gain coefficients $w_i(t)$ for the four pose classifiers depending on the frame index (t):

$$\mathbf{S} = \sum_{t=0}^T [\Pr_{pose_1}(t) \cdot w_1(t) + \Pr_{pose_2}(t) \cdot w_2(t) + \Pr_{pose_3}(t) \cdot w_3(t) + \Pr_{pose_4}(t) \cdot w_4(t)] \quad (12)$$

IV. RESULTS

A. Datasets

In order to evaluate and test the trained classifiers, two datasets were used. The search after best hyper-parameters of a single pose classifier will be performed by training and validating them on our **humiact5** dataset. Its consists of images of 5 two-person poses - snapshots of interactions: boxing, facing, hand holding, hand shaking and hugging/kissing. There are 1695 images in total, in which 1154 images are in the training set and remaining 541 images are in the evaluation set (Figure 7). In this series of experiments, the OpenPose library has been applied for skeleton detection in RGB images.

The best configuration of the pose experts and the final, time-accumulating network will be trained and tested on the interaction subset of the **NTU RGB+D** dataset. It includes 11 two-person interactions of 40 actors: A50: punch/slap, A51: kicking, A52: pushing, A53: pat on back, A54: point finger, A55: hugging, A56: giving object, A57: touch pocket, A58: shaking hands, A59: walking towards, A60: walking apart. In our experiments, already the skeleton data of the NTU RGB+D dataset is considered. There are 10420 video clips in total, in which ca. 70% are in the training set and remaining 30% are in the test set. No distinct validation subset is distinguished.

The NTU RGB+D dataset allows to perform a cross-subject (person) (short: CS) or a cross-view (CV) evaluation. In the cross-subject setting, samples used for training show actions performed by half of the actors, while test samples show actions of remaining actors. In the cross-view setting, samples recorded by two cameras are used for training, while samples recorded by the remaining camera - for testing. We apply the cross-subject (CS) evaluation mode, i.e., videos of 20 persons are used for training and videos of remaining 20 persons - for testing. The training set contains video clips of users identified as: 1, 2, 4, 5, 8, 9, 13, 14, 15, 16, 17, 18, 19, 25, 27, 28, 31,

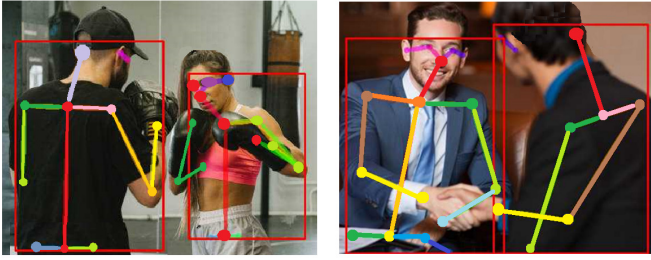


Fig. 7. Samples from our *humiact5* dataset: RGB images with skeleton data

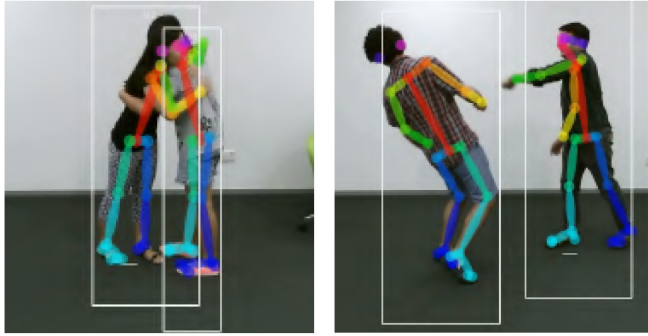


Fig. 8. Samples from the NTU RGB+D interaction dataset: RGB video frames with skeleton data [14]

34, 35 and 38. The number of samples in the training set is 7649, while in the test set - 2771.

Each skeleton instance consists of 25 joints of 3D skeletons that apparently represent a single person (Figure 8). As our research objective is to analyse video data and to focus on only reliably detected joints, we use only the 2D information of only first 15 joints.

From a video sample a set of frames is chosen as follows: the video clip is uniformly split into $N = 4$ time intervals ("periods"), from every interval some number of frames m is selected (we tested $m = 2, 4, 8$). The number of frames in the training set grows from 61192 to 244768 and the number of frames in the test set grows from 22168 to 88672, accordingly to the value of m from 2 to 8.

B. Pose classifier optimization

The hyper-parameter optimization of a pose classifier is performed on the small *humiact5* dataset. In order to run the *RandomSearch* function of Keras, a *NNHyperModel* is created, which implements the *HyperModel* class from the Keras-tuner. The hyper-parameters of the search space are declared in *NNHyperModel* as class parameters. Using the *RandomSearch* function, we identified three ANN configurations, each one being optimal for given number of hidden layers (1, 2 or 3).

The performances of the three selected models after 100 epochs of training are shown in Table I. The best test accuracy (i.e., the **recall** averaged over all classes) of 84% was achieved by the second model, whereas the other two have shown an accuracy of 82%. Consequently, we have chosen an ANN

TABLE I
THE MEAN ACCURACY ON THE *humiact5* DATASET OF THREE OPTIMAL ANN CONFIGURATIONS WITH 1, 2 AND 3 HIDDEN LAYERS.

Training/test	1 hidden	2 hidden	3 hidden
ANN - mean training accuracy	95%	96%	99%
ANN - mean test accuracy	82%	84%	82%

TABLE II
THE MEAN ACCURACY OF POSE CLASSIFIERS VERIFIED ON THE NTU RGB+D INTERACTION DATASET IN THE CS (CROSS SUBJECT) MODE

Expert	2 f/p	4 f/p	8 f/p
Pose - mean training accuracy	79.2%	82.4%	88.2%
Pose - mean test accuracy	61.2%	70.8%	76.1%

configuration of 2 hidden layers with 700 and 500 neurons in the first and second layer, respectively. The activation functions are ReLU and sigmoid, respectively. The learning rate is $5.89 \cdot 10^{-5}$.

C. Verification on the NTU RGB+D dataset

We train and test our models in the CS (*cross-subject*) verification mode proposed for the NTU RGB+D dataset, i.e. when actors in the training set are different than in the test set, but data from all the camera views are included in both sets. The frame sampling process for both training and testing will be done three times with different number of frames per time period (i.e., extracted from a single video sample): 2, 4, 8. The training set is split into learning and test subsets - two third for learning and one third for validation/testing. There are run 100 epochs of training and the best validation result will be chosen.

1) *Pose classifiers*: In the following, we apply the second version of the ANN pose classifiers, with two hidden layers, as reported earlier in Table I. We train four pose classifiers three times - every one is effectively trained on different frames according to its dedicated time period of action (i.e. $t \in [0, T/4], [T/4, 2T/4], [2T/4, 3T/4], [3T/4, T]$) of training samples with different frame sampling rates (i.e. $n = 2, 4, 8$ frames/period). The mean accuracy of these four pose experts, depending on the number of frames per period is shown on (Table II).

An immediate observation is, that all learning and test accuracies increase, when the training data size is increased. Specifically, with 8 frames per time-period (f/p), these accuracies reach to 88% and 76%, respectively. The average per class accuracies (i.e. four class poses representing the same interaction class) of the ANN experts, obtained with a 4 f/p

TABLE III
THE PER-CLASS TEST ACCURACY OF ANN POSE EXPERTS TRAINED ON THE NTU RGB+D INTERACTION DATASET, VERIFIED IN THE CS (CROSS SUBJECT) MODE, WHEN SAMPLED WITH 4 F/P

Class	A050	A051	A052	A053	A054	A055
Test accuracy	58%	52%	66%	69%	72%	83%
Class	A056	A057	A058	A059	A060	
Test accuracy	70%	64%	80%	81%	78%	

TABLE IV

THE ACCURACIES OF ANN POSE CLASSIFIER AND TWO VERSIONS OF THE ENSEMBLE CLASSIFIER (E-ANN-1, E-ANN-2), VERIFIED ON THE NTU RGB+D INTERACTION DATASET IN THE CS (CROSS SUBJECT) MODE

Classifier	Training accuracy	Test accuracy
Mean of pose classifiers	88.2%	76.1%
E-ANN-1, eq. (11)	92.4%	81.3%
E-ANN-2, eq. (12)	94.5%	83.3%

TABLE V

THE PER-CLASS TEST ACCURACY OF ANN ENSEMBLE CLASSIFIER, TRAINED ON THE NTU RGB+D INTERACTION DATASET, VERIFIED IN THE CS (CROSS SUBJECT) MODE, WHEN SAMPLED WITH 8 f/P

Class	A050	A051	A052	A053	A054	A055
Test accuracy	67%	67%	77%	87%	86%	91%
Class	A056	A057	A058	A059	A060	
Test accuracy	81%	76%	92%	93%	90%	

frame sampling on the test set, is shown on Table III. There are 3 classes (A55, A58, A59) that perform at least at 80%, other 6 classes - from 60% to 80% and two - below 60%. Compared with random choice - there are 11 classes and the random prediction (a guess) would be $1/11 = 9.09\%$. The largest accuracy is observed for the "A055 - hugging" class. The distance between two persons is here significantly smaller than of the rest and the poses are relatively stable in every time period.

2) *Ensemble of pose classifiers*: There are two variants of the final ensemble classifiers: E-ANN-1, when the final score of every interaction class is obtained by fixed weights, according to equation (11), or E-ANN-2, where the trainable gain network is used, according to equation (12). The class with highest score is selected as the winner of the interaction classifier. A notable improvement of interaction classification is observed, when accumulating over time sequence the weighted pose likelihoods. The mean accuracy of the best version of pose experts (i.e., for frame sampling of 8 f/p) was 88.2% (training) and 76.1% (testing), while the ensemble classifier has reached 92.4 % and 81.3% (version 1), or 94.5% and 83.3% (version 2), respectively (Table IV).

The per-class test accuracy of our ensemble classifier E-ANN-2 (with 8 f/p frame sampling) is shown in Table V. There are four classes (A55, A58, A59, A60) with an accuracy of 90% and higher, while the lowest performance (67%) is achieved for classes A50 (punch) and A51 (kicking). The confusion matrix for this testing case is shown in Figure 9. As the numbers of class samples in the test set are slightly unbalanced, we normalized the results, assuming 276 test instances per class, to make them easier comparable. "Punch" (A50) is most often misclassified with classes A51-A58, which all use hands to express an action, but most often is confused with "Point finger". "Kicking" (A51) is frequently confused with all other classes, slightly less with "pat on back". The main errors appear between actions "giving an object" (A56) and "shaking hands" (A58) - 18 and 9 cases, and between "pat on back" (A53) and "touch pocket" (A57) - 9 and 18 cases.

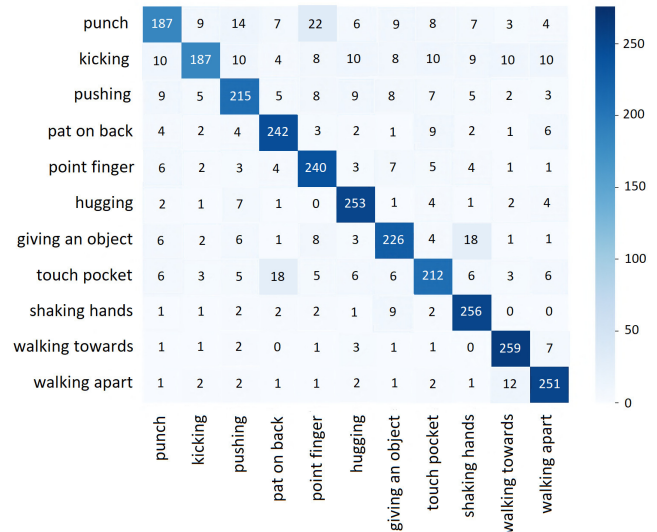


Fig. 9. The confusion matrix for the ensemble classifier E-ANN-2, verified on the NTU RGB+D interaction dataset in the CS (cross subject) mode

TABLE VI

INTERACTION CLASSIFICATION ACCURACY OF LEADING WORKS EVALUATED ON THE NTU RGB+D INTERACTION SET IN THE CS (CROSS SUBJECT) MODE. NOTE: † - RESULT ACCORDING TO [26], ‡ - RESULT ACCORDING TO [27]

Work - reference	Accuracy	Parameters	Frames
FSNET [32]	74.0% (†)	~ 200K	32
ST-LSTM [19]	83.0% (†)	~ 2.1M	32
ST-GCN [22]	83.3% (†)	3.08M	32
Our E-ANN-2	83.3%	400K	32
GCA-LSTM [33]	85.9% (†)	unknown	32
2S GCA-LSTM [34]	87.2% (†)	unknown	32
AS-GCN [23]	89.3% (†)	~ 9.5M	32
IRN _{inter+intra} [26]	85.4%	~ 9.0M	32
LSTM-IRN [26]	90.5%	~ 9.08M	max(all, 128)
2S-AGCN [24]	93.4% (‡)	3.0M	max(all, 300)
AAGCN [25]	91.5% (‡)	~ 6.0M	max(all, 300)
DR-GCN [27]	93.6%	3.18M	max(all, 300)
2S DR-AGCN [27]	94.6%	3.57M	max(all, 300)

D. Comparison

Many approaches to two-person interaction classification have been tested on the NTU RGB+D interaction dataset. We list some of the leading works in the Table VI. Our solution needs a low number of weights to be trained and it processes a sparse frame sequence. It shows a good tradeoff between competitive accuracy and low complexity when compared with other recently reported results.

Let us notice how we counted the number of parameters of the E-ANN-2 network. Remember that the pose classifiers have a common part - the feature transforming MLP with 2-hidden layers - and there are separate fully-connected output layers for every pose classifier. We can create two versions of the E-ANN network - one network with multiple feature-transforming MLPs that processes in parallel the four frame subsets, and another one that processes all frames in sequence.

As the individual results are finally aggregated over all frames, both configurations deliver the same final result. In the first configuration, there are 1 597 677 weights needed, while in the sequential version - 399 444 weights only:

- 1) The feature transforming ANN: $61 \cdot 700 + 700 + 700 \cdot 500 + 500 = 393\,900$ The FC classification layer: $500 \cdot 11 + 11 = 5\,511$ The gain network: $(11 + 11) + 11 = 33$
- 2) Four parallel pose classifiers: $4 \cdot 393\,900 + 4 \cdot 5\,511 + 33 = 1\,597\,677$
- 3) Four sequential pose classifiers: $393\,900 + 4 \cdot 5\,511 + 33 = 399\,444$

Taking into account, that the dominating processing time for a single frame is spent by the skeleton detector (on our equipment, it takes ca. 67 ms, compared to 1 ms for the pose classifier), the sequential version is preferred. Even when the skeleton detection itself will be performed in parallel, for every phase subset of frames one pose classifier will be allocated, the sequential version will take only $(N - 1)$ ms more time than when using N pose classifiers in parallel.

Typically, the performance of an interaction classifier is significantly improved when dense frame sequences are processed instead of sparse ones. But the overall processing time grows proportionally to the frame number, as the computation is dominated by the skeleton estimation step. Thus, processing a dense sequence of 100 frames (typical for the best performing solutions with accuracy $> 90\%$) takes roughly three times longer than the time needed for a sparse sequence of 32 frames (where a typical accuracy is $< 90\%$). The recently proposed multi-stream Graph CNNs have shown superior results but only when processing dense frame sequences. Considering the dominating processing time and resources needed for skeleton estimation in every frame of the sequence, the key to real-time interaction recognition is to limit the number of processed frames.

V. CONCLUSION

A light-weight approach to two-person interaction classification was proposed, that can be applied both in video- and single image-analysis. This is a skeleton-based approach, what means, that an external module for human detection and estimation in images is needed. We adopted the state-of-the-art OpenPose library for this purpose. This is a powerful deep network solution for human skeleton estimation in images. Our main contribution are algorithms for skeleton data correction and normalization and the design of an ANN classifier that has the form of an ensemble of several ANN-based pose experts. Aggregating four or more "weak" pose classifiers leads to an efficient and robust solution to human interaction classification. We also found that a comparison of classification approaches should not only consider the accuracy measure but also the amount of information received (i.e., whether a sparse or dense frame sequence is analyzed). Our future research should focus on the extraction of motion information for the skeleton joints and testing the model network on longer frame sequences.

REFERENCES

- [1] M. Liu and J. Yuan, "Recognizing Human Actions as the Evolution of Pose Estimation Maps", in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018*, Salt Lake City, UT, USA, June 18-22, 2018, pp. 1159-1168, doi: 10.1109/CVPR.2018.00127.
- [2] E. Cipitelli, E. Gambi, S. Spinsante, and F. Florez-Revue, "Evaluation of a skeleton-based method for human activity recognition on a large-scale RGB-D dataset," in *2nd IET International Conference on Technologies for Active and Assisted Living (TechAAL 2016)*, London, UK, 24-25 October 2016, pp. 1-6, doi: 10.1049/ic.2016.0063.
- [3] S. Zhang, Z. Wei, J. Nie, L. Huang, S. Wang, and Z. Li, "A Review on Human Activity Recognition Using Vision-Based Method," *Journal of Healthcare Engineering*, Hindawi, vol. 2017, Article ID 3090343, 31 pages, 2017, doi: 10.1155/2017/3090343, <https://www.hindawi.com/journals/jhe/2017/3090343/>
- [4] A. Wilkowska, W. Kasprzak and M. Stefanczyk, "Object detection in the police surveillance scenario," in *Proceedings of the 2019 Federated Conference on Computer Science and Information Systems, ACSIS*, vol. 18, 2019, pp. 363-372, doi: 10.15439/2019F291 .
- [5] A. Stergiou and R. Poppe, "Analyzing human-human interactions: A survey," *Computer Vision and Image Understanding*, Elsevier, vol. 188, 2019, p. 102799, doi: 10.1016/j.cviu.2019.102799, <https://www.sciencedirect.com/science/article/pii/S1077314219301158>
- [6] A. Bevilacqua, K. MacDonald, A. Rangrej, V. Widjaya, B. Caulfield, and T. Kechadi, "Human Activity Recognition with Convolutional Neural Networks," in *Machine Learning and Knowledge Discovery in Databases, ECML PKDD 2018*, Lecture Notes in Computer Science, vol. 11053, Springer, Cham, Switzerland, 2019, pp. 541-552, doi: 10.1007/978-3-030-10997-4_33.
- [7] N. A. Mac and N. H. Son, "Rotation Invariance in Graph Convolutional Networks," in *Proceedings of the 16th Conference on Computer Science and Intelligence Systems, ACSIS*, vol. 25, 2021, pp. 81-90, doi: 10.15439/2021F140 .
- [8] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Field," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 1, pp. 172-186, Jan. 2021, doi: 10.1109/TPAMI.2019.2929257.
- [9] A. Toshev and C. Szegedy, "DeepPose: Human Pose Estimation via Deep Neural Networks," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, 2014, pp. 1653-1660, doi: 10.1109/CVPR.2014.214.
- [10] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele, "Deepercut: a deeper, stronger, and faster multi-person pose estimation model," in *Computer Vision – ECCV 2016*, Lecture Notes in Computer Science, vol. 9910, Springer, Cham, Switzerland, 2016, pp. 34-50. https://doi.org/10.1007/978-3-319-46466-4_3.
- [11] H.-D. Duan, J. Wang, K. Chen and D. Lin, "PYSKL: Towards Good Practices for Skeleton Action Recognition," arXiv:2205.09443v1[cs.CV], 15 May 2022, <https://arxiv.org/abs/2205.09443v1> (accessed on 15.07.2022).
- [12] [Online], "Papers with code. Action recognition in videos," <https://paperswithcode.com/task/action-recognition-in-videos>, (accessed on 15.07.2022).
- [13] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, "Adaptive mixtures of local experts", *Neural Computation*, vol. 3, no. 1, pp. 79-87, March 1991, doi: 10.1162/neco.1991.3.1.79.
- [14] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "NTU RGB+D: A Large Scale Dataset for 3D Human Activity Analysis," arXiv:1604.02808[cs.CV], 2016, <https://arxiv.org/abs/1604.02808> (accessed on 15.07.2022).
- [15] H. Meng, M. Freeman, N. Pears, and C. Bailey, "Real-time human action recognition on an embedded, reconfigurable video processing architecture," *J. Real-Time Image Proc.*, vol. 3, no. 3, pp. 163-176, 2008, doi: 10.1007/s11554-008-0073-1.
- [16] K.G. Manosha Chathuramali and R. Rodrigo, "Faster human activity recognition with SVM," *International Conference on Advances in ICT for Emerging Regions (ICTer2012)*, Colombo, Sri Lanka, 12-15 December 2012, IEEE, 2012, pp. 197-203, doi: 10.1109/ictcr.2012.6421415.
- [17] X. Yan and Y. Luo, "Recognizing human actions using a new descriptor based on spatial-temporal interest points and weighted-output classifier," *Neurocomputing*, Elsevier, vol. 87, pp. 51-61, 15 June 2012, doi: 10.1016/j.neucom.2012.02.002.

- [18] R. Vemulapalli, F. Arrate, and R. Chellappa, "Human Action Recognition by Representing 3D Skeletons as Points in a Lie Group," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 23-28 June 2014, Columbus, OH, USA, IEEE, 2014, pp. 588-595, doi: 10.1109/cvpr.2014.82.
- [19] J. Liu, A. Shahroudy, D. Xu, and G. Wang, "Spatio-Temporal LSTM with Trust Gates for 3D Human Action Recognition," in *Computer Vision – ECCV 2016, Lecture Notes in Computer Science*, vol. 9907, Springer, Cham, Switzerland, 2016, pp. 816–833, doi: 10.1007/978-3-319-46487-9_50.
- [20] C. Li, Q. Zhong, D. Xie, and S. Pu, "Skeleton-based Action Recognition with Convolutional Neural Networks," *2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, 10-14 July 2017, Hong Kong, pp. 597-600, doi: 10.1109/ICMEW.2017.8026285.
- [21] D. Liang, G. Fan, G. Lin, W. Chen, X. Pan, and H. Zhu, "Three-Stream Convolutional Neural Network With Multi-Task and Ensemble Learning for 3D Action Recognition," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 16-17 June 2019, Long Beach, CA, USA, IEEE, pp. 934-940, doi: 10.1109/cvprw.2019.00123.
- [22] S. Yan, Y. Xiong, and D. Lin, "Spatial Temporal Graph Convolutional Networks for Skeleton-Based Action Recognition," arXiv:1801.07455 [cs.CV], 2018, <https://arxiv.org/abs/1801.07455>, (accessed on 15.07.2022).
- [23] M. Li, S. Chen, X. Chen, Y. Zhang, Y. Wang, and Q. Tian, "Actional-Structural Graph Convolutional Networks for Skeleton-Based Action Recognition," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, 15-20 June 2019, pp. 3590-3598, doi: 10.1109/CVPR.2019.00371.
- [24] L. Shi, Y. Zhang, J. Cheng and H.-Q. Lu, "Two-Stream Adaptive Graph Convolutional Networks for Skeleton-Based Action Recognition," arXiv:1805.07694v3 [cs.CV] , 10 July 2019, doi: 10.48550/ARXIV.1805.07694, <https://arxiv.org/abs/1805.07694v3>, (accessed on 15.07.2022).
- [25] L. Shi, Y. Zhang, J. Cheng, and H.-Q. Lu, "Skeleton-based action recognition with multi-stream adaptive graph convolutional networks," *IEEE Transactions on Image Processing*, vol. 29, pp. 9532-9545, October 2020, doi: 10.1109/TIP.2020.3028207 .
- [26] M. Perez, J. Liu, and A.C. Kot, "Interaction Relational Network for Mutual Action Recognition," arXiv:1910.04963 [cs.CV], 2019, <https://arxiv.org/abs/1910.04963> (accessed on 15.07.2022).
- [27] L.-P. Zhu, B. Wan, C.-Y. Li, G. Tian, Y. Hou and K. Yuan, "Dyadic relational graph convolutional networks for skeleton-based human interaction recognition," *Pattern Recognition*, Elsevier, vol. 115, 2021, p. 107920, doi: 10.1016/j.patcog.2021.107920.
- [28] [Online], "openpose", CMU-Perceptual-Computing-Lab, 2021 <https://github.com/CMU-Perceptual-Computing-Lab/openpose/> , (accessed on 15.07.2022).
- [29] [Online], "Keras: the Python deep learning API," <https://keras.io/> , (accessed on 15.07.2022).
- [30] [Online], "Keras Tuner," <https://keras-team.github.io/keras-tuner/> , (accessed on 15.07.2022).
- [31] T. Yu and H. Zhu, "Hyper-Parameter Optimization: A Review of Algorithms and Applications," arXiv:2003.05689 [cs.LG], 12 Mar 2020, <https://arxiv.org/abs/2003.05689> , (accessed on 15.07.2022).
- [32] J. Liu, A. Shahroudy, G. Wang, L.-Y. Duan, and A. C. Kot, "Skeleton-Based Online Action Prediction Using Scale Selection Network," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 42, no. 6, pp. 1453–1467, 1 June 2020, doi: 10.1109/TPAMI.2019.2898954.
- [33] J. Liu, G. Wang, P. Hu, L.-Y. Duan, and A. C. Kot, "Global Context-Aware Attention LSTM Networks for 3D Action Recognition," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 21-26 July 2017, pp. 3671-3680, doi: 10.1109/CVPR.2017.391.
- [34] J. Liu, G. Wang, L.-Y. Duan, K. Abdiyeva, and A. C. Kot, "Skeleton-Based Human Action Recognition with Global Context-Aware Attention LSTM Networks," *IEEE Transactions on Image Processing (TIP)*, vol. 27, no. 4, pp. 1586-1599, April 2018, doi: 10.1109/TIP.2017.2785279.

Geometry-Aware Keypoint Network: Accurate Prediction of Point Features in Challenging Scenario

Tomasz Nowak

Institute of Robotics and Machine Intelligence
 Poznań University of Technology
 ul. Piotrowo 3A, 60-965 Poznań, Poland
 Email: tomasz.nowak@doctorate.put.poznan.pl

Piotr Skrzypczyński

Institute of Robotics and Machine Intelligence
 Poznań University of Technology
 ul. Piotrowo 3A, 60-965 Poznań, Poland
 Email: piotr.skrzypczynski@put.poznan.pl

Abstract—In this paper, we consider a challenging scenario of localising a camera with respect to a charging station for electric buses. In this application, we face a number of problems, including a substantial scale change as the bus approaches the station, and the need to detect keypoints on a weakly textured object in a wide range of lighting and weather conditions. Therefore, we use a deep convolutional neural network to detect the features, while retaining a conventional procedure for pose estimation with 2D-to-3D associations. We leverage here the backbone of HRNet, a state-of-the-art network used for detection of feature points in human pose recognition, and we further improve the solution adding constraints that stem from the known scene geometry. We incorporate the reprojection-based geometric priors in a novel loss function for HRNet training and use the object geometry to construct sanity checks in post-processing. Moreover, we demonstrate that our Geometry-Aware Keypoint Network yields feasible estimates of the geometric uncertainty of point features. The proposed architecture and solutions are tested on a large dataset of images and trajectories collected with a real city bus and charging station under varying environmental conditions.

I. INTRODUCTION

Estimation of the absolute pose of a camera from a single image is a classic problem in computer vision [1], which is being solved in a number of ways, depending on the application. Conventional, structure-based algorithms yield accurate camera pose estimates if stable, salient point features can be extracted. Unfortunately, in a number of practical scenarios, particularly outdoors, good quality point features (keypoints) are hard to extract from images, due to weak textures, difficult lighting, and motion blur.

The localisation scenario considered in this paper is part of the Advanced Driver Assistance System (ADAS) which we developed in cooperation with Solaris Bus & Coach, one of the manufacturers of electric buses [2]. The ADAS provides a bus driver with visual cues on how to operate the steering wheel in order to perform the desired manoeuvre. To perform successful docking, the driver has to put the tip of the vehicle's pantograph into the head of the charger, which is mounted on a support pylon. An important prerequisite for successfully planning such a manoeuvre is an accurate estimate of the bus position in relation to the charger head and its pylon. The use of GPS for localising while docking is considered unreliable in an urban environment. Active beacons or even large passive



Fig. 1. Accurate detection of keypoint features for vision-based localisation of an electric bus while docking to a charging station. The inset images show a charging station with the ground truth keypoints (upper), the estimated locations of these keypoints with uncertainty ellipses (lower), and a close-up of the roof-mounted sensor unit with a camera (in the circle)

markers installed at the charging station cannot be considered as well, because bus operators often do not permit deployment of any additional elements at the stations due to legal issues. For localisation, the bus has only a monocular camera mounted in the front part of the roof. The use of such a simple sensor was required by the bus manufacturer, as the ADAS equipment has to be affordable and scalable to different bus models.

The charging station is detected automatically from a long distance (typically 30 m), using the approach introduced in [3]. Once the station gets detected, the task comes down to the estimation of the camera pose with respect to certain predefined points of the charger structure. Despite its apparent simplicity, this scenario raises a number of issues in the camera pose estimation method. Firstly, the method needs to work without an initial pose guess, considering each observation of the charging station as a separate global localisation act. Secondly, the charging station is assumed to be the only known object in the environment, for which we have a 3-D model. Hence, we need to use the keypoints defined on the charging station that are observed during the entire docking manoeuvre over a large range of viewpoints, appearance, and scale change. All these difficulties make the existing simultaneous localisation

and mapping (SLAM) or visual odometry (VO) algorithms impractical in our scenario, as SLAM and VO systems need to be initialised with a known camera pose and require the salient features to be present in the environment all along the executed trajectory.

To address the specific requirements we proposed in [4] a two-step procedure, which uses a conventional, structure-based pose estimation algorithm with 2D-to-3D associations between the keypoints found in the camera image and predefined points from the three-dimensional model of the charger. Whereas this procedure using a Faster R-CNN network architecture adopted to detect the keypoints has been positively verified in docking experiments with a real bus [2], [4], we observed, that any inaccuracy in the position of the estimated keypoints significantly deteriorates the accuracy of the final position estimate. A mismatch of detected points with other features makes the pose estimate completely wrong.

Therefore, in this paper we investigate how to adapt to our application a different network architecture: the High Resolution Network (HRNet) [5], which is a leading solution for keypoint detection in human pose estimation. The detection of body points in humans is a major area of interest for researchers and commercial use, and thus it arguably sets the state-of-the-art in keypoint detection in the wild [6]. We conjecture that a network architecture achieving top scores in the COCO Keypoint Detection Task would be a good starting point for use in the considered application. However, our application offers some a priori knowledge about the geometry of the observed object, which is not present in the human pose estimation task. Thus, we investigate how to include this knowledge either in the learning process, creating an inductive bias in the neural network, or in post-processing, defining a sanity check procedure that quickly eliminates implausible predictions of the network. We also extend the neural network architecture by adding a separate branch that estimates covariance matrices describing the 2-D uncertainty of the detected keypoints. Finally, we get the new Geometry-Aware Keypoint Network that addresses the specific challenges of the bus localisation process at the charging station.

In this work, we contribute: (i) an analysis of the HRNet architecture aimed at the accuracy improvement of the keypoints locations with respect to ground truth, also considering the computation burden; (ii) a novel loss function that exploits the available geometric priors of our application; (iii) a sanity check procedure based on these geometric priors, and (iv) an extended experimental evaluation of all these components on a unique application-specific dataset, which is made publicly available.

The remainder of this paper is organised as follows. Section II reviews the most important related work in similar applications and in keypoints detection from monocular images. Section III gives a description of the proposed neural network architecture, and provides technical details pertaining to its novel components. Then, section IV describes the experimental procedure and summarises the results of the experiments, while section V draws conclusions upon these results.

II. RELATED WORK

A. Vision for automated docking

There are few works concerning vision-guided docking of larger vehicles to electric chargers [7]. Precise docking to a charging station under the control of a camera can be cast as a visual servoing problem. Unfortunately, visual servoing methods [8] require the target object or marker to appear big enough in the images, whereas the charging station detected from a distance of 30 m is too small to make a visual servoing method effective. Recent visual SLAM algorithms can estimate a camera's trajectory precisely over hundreds of metres [9]. In practice, however, a SLAM method requires a large number of features detected over several consecutive image frames [10] and has to be initialised properly, which is problematic in the monocular case and often requires several attempts with camera relocation. These problems make SLAM impractical for the specific task we consider, as neither visual odometry nor SLAM exploit the knowledge about the appearance and geometry of the charging station, which is assumed to be visible during the entire docking manoeuvre.

On the other hand, automated charging for self-driving electric cars is often implemented using devices that plug into the car's charging port [11]. This approach eases the guidance process, as only this port has to be localised with respect to the plugging arm/device. In this context, the approach we propose is of practical importance, as we do not need any active devices, nor markers/fiducials attached to the charging station, and we can localise the bus during the entire manoeuvre, starting 30 metres from the station.

B. Camera pose estimation

A wide variety of algorithms have been presented in the literature for the camera pose estimation problem. The classic way is to compute the camera pose from 2D-to-3D correspondences between 2-D features in the image and 3-D points in the model [1]. The model can be given a priori, from CAD data or via accurate laser scanning, as in our scenario, or can be obtained through 3-D structure-from-motion (SfM). Correspondences between points are established through the matching of descriptors, which can either be handcrafted [12] or learned [13]. The camera pose is then computed upon the known correspondences applying a variant of the perspective- n -point (PnP) algorithm [14] or optimised using bundle adjustment [15]. The conventional approach can estimate the camera pose accurately [16], but it struggles whenever the features are difficult to match. Our approach follows the classic pipeline with respect to pose computation using a PnP algorithm, but deploys a specialised neural network as the feature detector, thus obtaining a well-defined pattern of a few keypoints that are already associated with the model points. Owing to this concept we do not need to use RANSAC for outlier rejection.

In the last few years, learning-based approaches to camera pose estimation have been gaining attention. End-to-end methods have been pioneered by PoseNet [17], which used a trained convolutional neural network to regress a six degrees-of-freedom camera pose. A similar idea was followed by

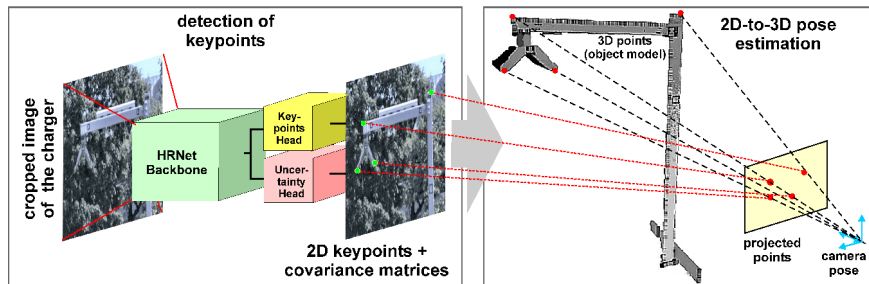


Fig. 2. Structure of the camera pose estimation system consisting of a CNN-based keypoint detection module and a conventional pose computation method

[18], demonstrating camera relocalisation in cluttered scenes. Although PoseNet can localise the camera without an initial pose guess, as we require in our scenario, the accuracy of pose estimates yielded by this neural network is inferior with respect to its conventional, structure-based counterparts using principled algorithms. As concluded in [19], the main reason for the insufficient accuracy was the use of a naive regression loss function without consideration of the scene geometry. The improved PoseNet presented in [19] uses reprojection-based loss, thus narrowing the performance gap to pose estimation using principled algorithms. The importance of using the geometric knowledge about the scene and enforcing the geometric constraints during learning was pointed out in [20], and has been demonstrated recently also by other pose estimation methods. A camera pose estimation pipeline that is applicable for fusing classical geometry and deep learning is proposed in [21], while [22] introduces an end-to-end system consisting of deep learning modules for feature extraction, matching, and outlier rejection while optimising for the camera pose objective.

C. Detection of keypoints

In conventional, structure-based pose estimation systems detectors and descriptors designed by heuristics are applied. However, in recent years, deep learning has been applied to obtain feature detectors and descriptors in visual odometry and SLAM. LIFT [23] was among the first attempts to detect features by end-to-end learning, and was trained on ground truth obtained from SIFT and SfM. The more recent SuperPoint [13] introduced a self-supervised approach to learn the detectors and descriptors of keypoints simultaneously, while DISK [24] demonstrated learning of features using policy gradient.

Independently, a number of neural network architectures for keypoint detection have been proposed in the context of human pose estimation. The architecture from [25] is based on Faster R-CNN with ResNet-101 backbone, similarly to our previous camera pose estimation system described in [4]. Whereas this approach yields accurate keypoints, and thus is included in our experiments for comparison, it is computationally heavy and hence does not allow to exploit the full resolution of the acquired images. A recent paper [26] demonstrated the use of novel self-calibrated convolutions that expand fields-of-view of each convolutional layer to detect keypoints. A leading

solution for keypoint detection in human pose estimation is the High Resolution Network [5], which was designed specifically to keep the high resolution of the feature maps through the entire processing pipeline. This approach is in line with our idea of using high-resolution images for reliable detection of keypoints from longer distances, while still keeping good accuracy of their localisation in images. Therefore, we selected the HRNet as our baseline approach and adopted its backbone network as a feature extractor in our GAKN architecture.

Uncertainty in learned feature extractors and camera pose estimation methods seems to be not yet fully exploited, despite its importance in the conventional, geometric approaches. Among the examples considering uncertainty, [27] uses Bayesian neural networks to obtain localisation uncertainty, and a recent approach [21] learns the deep neural network uncertainty guided by the geometric uncertainty. In our localisation scenario, we are interested in aleatoric uncertainty, which depends on the inputs, and may be estimated from data [28]. Whereas Bayesian deep learning is popular for this purpose, we leverage the Cholesky Estimator Network to represent the uncertainty of each keypoint as a Gaussian distribution with covariance matrix [29].

III. STRUCTURE OF THE PROPOSED SOLUTION

The main inspiration for this work was an excellent performance of state-of-the-art keypoint detectors in the top down human pose estimation methods [6]. In the top down approach to human pose estimation keypoint locations are predicted within bounding boxes obtained from a person detector. This fits our processing pipeline, where the charging station is detected by a Faster R-CNN object detector [3], and further processing for localisation is limited to the image area cropped by the object's bounding box [4] (Fig. 2).

Therefore, we adopt the keypoint detection architecture with HRNet [5] backbone implemented using the MMPose framework [30] as a baseline model during our experiments. This architecture consists of a backbone network and the keypoint head. We decided to use a pre-trained backbone, and design our own head, also adding an auxiliary branch for the estimation of spatial uncertainty of the keypoints (Fig. 3). The keypoint head contains Deconv Blocks which double the resolution of the feature maps. A single Deconv Block is built with a Transposed Convolution Layer followed by a Batch

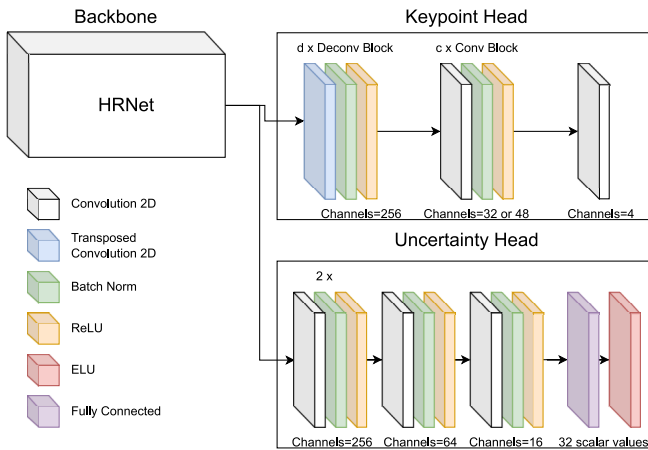


Fig. 3. Architecture of the GAKN model. Configurations with heatmap size of 128×128 , 256×256 and 512×512 contains d equal 0, 1 or 2 Deconv Blocks respectively. Architectures can be extended with c additional Conv Blocks in the keypoint head

Norm and ReLU layers. The final layer in the keypoint head is a single Convolution Layer that outputs n heatmaps, where n is the number of keypoints. For the preparation of ground truth heatmaps as training targets and the decoding of the final keypoint coordinates from the heatmaps we incorporated the UDP and DarkPose methods described in [31] and [32], respectively. These data preprocessing techniques allow us to preserve the accuracy of ground truth coordinates of the keypoints through the labelling and augmentation process. During the inference, while decoding the coordinates from predicted heatmaps, those methods allow for extracting unbiased, subpixel-accuracy keypoint locations.

Point detection in objects such as an electric bus charger is different from body point detection in humans. The human body is composed of numerous parts moving relatively to each other, therefore choosing which part of the image should be used to look for a given point is not obvious. Due to the rigid geometric configuration of the charging station, this is not a problem in our network design, thus we focused on developing methods to increase the accuracy of point localisation. We experimented with several configurations of the HRNet architecture in order to find the limits of the keypoints location accuracy, and to choose a reasonable trade-off between the accuracy and the number of computations. We selected three aspects of the HRNet model for potential improvements: the selection of a backbone network for feature extraction, the size of the returned heatmaps, and the structure of the keypoint head. Next, having a reasonable baseline design customised to our application, we investigated the best way of injecting the prior geometric knowledge to the network model.

A. Backbone network

We evaluated two backbone networks HRNet32 and HRNet48. The difference between HRNet48 and HRNet32 lies

in the width of the convolutional layers in the high-resolution stream (48 and 32 respectively). This enhances the capability of HRNet48 to extract more feature maps but at the cost of higher computational cost.

B. Keypoint head depth

We evaluated the influence of the additional convolutional layers in the keypoint head to find out whether it will improve the pose estimation accuracy. The extra layers were added after the Deconv Block and before the convolution layer which produces final heatmaps. Each extra layer consists of 256 filters of the size 3×3 and is followed by Batch Norm and ReLU.

C. Heatmap size

The default implementation of the keypoint detector based on HRNet returns heatmaps which are downsampled four times compared to the input image size, so using an image of 512×512 pixels results in 128×128 pixels heatmaps. The upsampling of the heatmaps is achieved using Transposed Convolutional layers. A single transposed convolutional layer increases the width and height of the heatmap twice.

The charging station pylon and head do not contain moving parts and their geometric configuration is constant. This makes rough estimation of the keypoints locations easier than the same task in human pose estimation because we can expect the given point in the specific area of the image. On the other hand, the accuracy of the described pose estimation method is strongly dependent on the accuracy of estimation of the keypoints. Small inaccuracies in the location of keypoints propagate to relatively large pose estimation errors, especially from larger distances. The above considerations suggest, that increasing the resolution of the output heatmaps may promote accurate subpixel estimation of the keypoint locations, which in turn will lead to a significant improvement of the camera pose estimation accuracy.

D. Reprojection loss

The key role of scene geometry in conventional pose estimation models and the fact that ground truth points can be easily identified for the charging station, together with the geometric relations between these points, were inspirations for developing the HRNet baseline model into the Geometry-Aware Keypoint Network.

This network exploits these geometric priors while training, as it uses an additional cost function based on the reprojection error. The reprojection loss penalises spatial configurations of the keypoints which are physically impossible. We define camera projection function, π , which maps the i -th 3-D point \mathbf{w}_i to the 2-D image point $(\tilde{u}, \tilde{v})^T$ leveraging the given camera intrinsics parameters \mathbf{K} :

$$\pi(\mathbf{T}, \mathbf{K}, \mathbf{w}_i) \mapsto (\tilde{u}, \tilde{v})^T, \quad (1)$$

where \mathbf{T} is a rigid transformation matrix (rotation and translation).

To calculate the reprojection loss, we minimise the difference between the projection of the real 3-D object points $(\tilde{u}, \tilde{v})^T$ and the points predicted by the network $(\hat{u}, \hat{v})^T$. The optimisation problem is solved by the Trust Region Reflective algorithm [33] which finds a transformation \mathbf{T}^* :

$$\mathbf{T}^* = \underset{\mathbf{T}}{\operatorname{argmin}} \sum_{i=1}^n (\tilde{u}_i - \hat{u}_i)^2 + (\tilde{v}_i - \hat{v}_i)^2. \quad (2)$$

Trust Region Reflective is a robust optimisation method for constrained problems that has a Python implementation in the SciPy library, which facilitates integration in a deep learning framework.

To limit the search space and keep the solution physically correct we applied constraints on the transformation \mathbf{T} to reflect only the operating area of the bus. All three values of the rotation vector are limited to $\frac{\pi}{4}$. Assuming that roll and pitch angle are close to zero, this constraint limits the yaw angle to be less than $\pm \frac{\pi}{4}$. The translation in the lateral axis is limited to ± 20 m, the longitudinal axis is limited to 50 m and the translation in the z axis is limited to 50 m (Fig. 4).

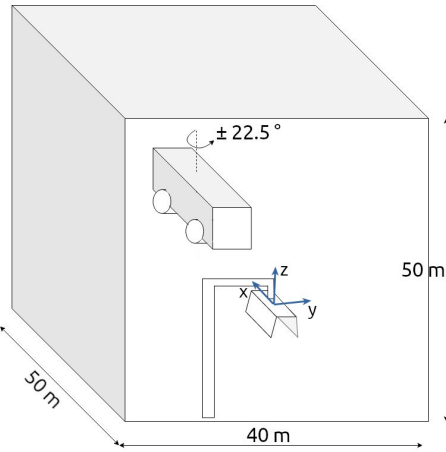


Fig. 4. The considered search space for the bus location and orientation.

Then, the reprojection loss is the value of the cost function for the optimal \mathbf{T}^* transformation:

$$\operatorname{loss}_{\text{repr}} = \sum_{i=1}^n (\pi(\mathbf{T}^*, \mathbf{K}, w_i^x) - \hat{u}_i)^2 + (\pi(\mathbf{T}^*, \mathbf{K}, w_i^y) - \hat{v}_i)^2, \quad (3)$$

where w_i^x and w_i^y are x and y coordinates of point \mathbf{w}_i .

The second loss element is a Mean Squared Error Loss:

$$\operatorname{loss}_{\text{MSE}} = \sum_{i=1}^n \left(\frac{1}{m} \sum_{j=1}^m (m_{ij} - \hat{m}_{ij})^2 \right), \quad (4)$$

where m_{ij} is the j -th pixel of the ground truth heatmap corresponding to the i -th point and \hat{m}_{ij} is the j -th pixel of the predicted heatmap corresponding to the i -th point. The final loss function is a sum of the MSE loss (4) and the reprojection loss, calculated according to the formula:

$$\operatorname{loss} = \operatorname{loss}_{\text{MSE}} + \lambda \cdot \operatorname{loss}_{\text{repr}}, \quad (5)$$

where the metaparameter λ was estimated experimentally and is set to 0.01 for our evaluation.

E. Reprojection-based refinement

Each of the four points defined on the charging station can be detected more or less accurately, depending on a number of circumstances, including the camera viewpoint and the amount of motion blur in the image. In this situation, it may happen, that some keypoints are well extracted, while some others are inaccurate or even misplaced.

To cope with this problem we define a sanity check procedure that incorporates the geometric constraints during the inference of the neural network. This procedure refines the neural network predictions, solving the same task as described by the equation (2). Having the optimal transformation \mathbf{T}^* , we project 3-D coordinates of the keypoints on the image. Then the distances between the predicted point $(\hat{u}, \hat{v})^T$ and the projected point $(\tilde{u}, \tilde{v})^T$ are calculated. Then, we compare the maximum distance d_{\max} with the mean of three remaining distances d_{res} multiplied by parameter $\gamma = 2$. If the inequality $d_{\max} > \gamma d_{\text{mean}}$ is satisfied, we use this projection as the final prediction of keypoints location. The above condition ensures that we are refining only the cases where there is only one point with inaccurate prediction.

F. Uncertainty estimation

As knowing the geometric uncertainty of extracted keypoints may be instrumental when computing the camera pose estimate and then fusing it with other localisation data in the vehicle, we extended the GAKN architecture with an uncertainty estimation branch. Our aim was to obtain a trainable model that predicts covariance matrices of the individual keypoints depending on the input images. With such a model we can judge if the extracted keypoints are accurate enough for the camera (and then vehicle) localisation task. For this purpose, we implemented the Gaussian Log-Likelihood Loss proposed in [29]. We estimate the uncertainty of all four keypoint locations as a Gaussian distribution with covariance matrix Σ , an 8×8 symmetric positive definite matrix. Σ has $(2n + 1)n$ degrees of freedom which are estimated by a lower triangular matrix \mathbf{L} such that $\Sigma = \mathbf{L}\mathbf{L}^T$ (Cholesky decomposition). The GAKN uncertainty estimation branch consists of four blocks built with a convolutional layer followed by Batch Norm and ReLU. The final layer is Fully Connected which predicts 36 values of the \mathbf{L} matrix. The loss function used during training is described by:

$$\operatorname{loss}_{\text{unc}} = \sum_{i=1}^n \log |\Sigma| + (\mathbf{p} - \hat{\mathbf{c}}) \Sigma^{-1} (\mathbf{p} - \hat{\mathbf{c}}), \quad (6)$$

where \mathbf{p} is a vector of ground truth keypoint locations and $\hat{\mathbf{c}}$ is a vector of predicted keypoint locations. The 2×2 covariance matrices of the individual keypoints are then extracted from Σ , and can be visualised as uncertainty ellipses in the image plane.

IV. EXPERIMENTS

A. Evaluation procedure

The purpose of the presented experiments was to examine the influence of the neural network architecture on the 2-D pose estimation accuracy and the computational complexity. All experiments were performed off-line using Nvidia A100 GPU on a custom dataset recorded using a real electric bus and charger with the ground truth poses obtained using Differential GPS (DGPS). In our experimental installation (cf. Fig. 1) a high-resolution camera (5472×3648 pixels) is used, as the application scenario requires to detect the charging station from a distance of at least 30 m and localise with a 2-D circular position error smaller than 0.35 m when approaching the charger’s head (this error can be compensated mechanically). The camera is mounted in a detachable unit with a laser scanner for safety monitoring (not used here), and a DGPS receiver, which was used only to obtain ground truth trajectories of the bus. Note that compared to [4] we no longer use the black rectangular markers that are visible on the charger’s pylon (cf. Fig. 1). The ground truth keypoints are defined by manual labelling directly on the training sequence images. They are two corners of the charger’s head and two other points on the extreme parts of the supporting pylon. The dataset consists of 81 image sequences gathered over five days. The diversity of data was achieved by different manoeuvre starting points, different bus trajectories, and changing lighting and weather conditions.

The final pose estimate of the camera is computed as in [4], using an iterative perspective- n -point algorithm, which minimises the reprojection error. The cost function of this optimisation problem is defined as sum of squared distances between the point localisation on the image, and the object model points projected on this image:

$$\mathbf{T}^* = \underset{\mathbf{T}}{\operatorname{argmin}} \sum_{i=1}^n \left((\hat{u}, \hat{v})_i^T, \pi(\mathbf{T}\mathbf{w}_i) \right)^T \left((\tilde{u}, \tilde{v})_i^T, \pi(\mathbf{T}\mathbf{w}_i) \right). \quad (7)$$

To consider the detection of a keypoint as accepted, the RMSE of the 3-D point projected on the image using ground truth transformation from DGPS should be less than d pixels:

$$\sqrt{\frac{1}{n} \sum_{i=1}^n \left(\sqrt{(\tilde{u} - \hat{u})^2 + (\tilde{v} - \hat{v})^2} \right)^2} < d, \quad (8)$$

with d set to 10 in our experiments.

Finally, we evaluated the performance of the different neural models applied for keypoints extraction directly for the objective of camera pose accuracy. The ground truth trajectories were provided by DGPS (2 cm of accuracy), while the camera and DGPS receiver were calibrated using an optimisation-based procedure developed specifically for this project [34]. The 3-D model of the charging station was obtained using a geodetic laser scanning system. For evaluation we use three metrics: median of the 2-D translation error, the median of the yaw angle estimation, and percentage of accepted detections. We project the 3-D camera pose into the 2-D manifold and

ignore the pitch and roll angles, as these dimensions are irrelevant to localisation of the bus on a flat area. For all experiments, the image input size was set to 512×512 pixels. Results of the main experiments are presented in Tab. I. The considered configurations of the keypoint extraction network are the Baseline (i.e. HRNet32 with 3 layers head), Refined, which is Baseline with the geometric sanity check, GAKN, and GAKN+Refined, which is GAKN equipped with the sanity check in the post-processing.

B. Comparison with the state-of-the-art

To compare the GAKN architecture with the state-of-the-art we use the HRNet configured for the considered application scenario according to the outcome of our experiments, i.e. with the HRNet32 backbone and the keypoint head with 3 layers, which we call the “Baseline” configuration. Additionally, we use for comparison the Faster R-CNN-based network described in [25]. During the inference, this network required above 32 GB of GPU memory, most of it was consumed by the extraction of the keypoint location offset values. The inference time is significantly longer and the pose estimation errors are larger than for the HRNet-based networks. In our application scenario, the weakest point of the network from [25] is a low percentage of accepted detections (40.9 %). To improve this parameter, we tested also our sanity check procedure (post-processing) with this network architecture. However, in this case, application of the reprojection-based refinement does not bring the expected results. The ratio of accepted detections increased to above 50 % but the pose estimation accuracy dropped. This case shows that the reprojection-based refinement can improve results only when the raw predictions are relatively close to ground-truth and there is usually only a single keypoint of lower accuracy.

C. Backbone network

Comparing the HRNet32 and the HRNet48 (Tab. II) we can notice that the medians of rotation and translation errors are lower for the HRNet48 network for both keypoint head variants. There is no significant influence on the percentage of accepted detections. The inference time is slightly longer for the HRNet48 version and the required operations and number of parameters are over twice as large as for the HRNet32 backbone. Using the HRNet48-based network, the whole processing pipeline requires more than 8 GB of GPU memory. The commercial application of this system requires that the cost of used hardware must be considered. Despite better pose estimation accuracy, we selected the HRNet32 backbone for the Baseline, as it provides acceptable accuracy and can be run on low-end hardware (GPU). Consequently, the GAKN architecture configurations evaluated in our experiments also use the HRNet32 backbone.

D. Keypoint head depth

Additional convolutional layers in the keypoint head decrease slightly the median of translation and rotation error (Tab. II). Applying this modification does not affect the

TABLE I

COMPARISON OF THE SIZE OF NETWORKS, INFERENCE TIME, POSE ESTIMATION ERRORS AND PERCENT OF ACCEPTED DETECTIONS OF THE EVALUATED ARCHITECTURES.

Heatmap size	Configuration	Operations [GFLOPs]	Parameters [M]	Inference time [ms]	Median t _{2D} [m]	Median r _{2D} [deg]	Percent of accepted detections
152×152	Papandreou	41.22	42.67	167.82	0.53	1.15	40.90 %
	Papandreou+Refined			171.82	1.16	1.67	52.34 %
128×128	Baseline	41.08	28.54	35.60	0.43	0.97	92.59 %
	Refined			40.60	0.44	0.98	95.66 %
	GAKN			35.60	0.37	0.67	92.14 %
	GAKN+Refined			40.60	0.37	0.67	94.79 %
256×256	Baseline	43.34	28.67	38.40	0.35	0.74	93.56 %
	Refined			42.40	0.36	0.74	95.99 %
	GAKN			38.40	0.32	0.62	95.56 %
	GAKN+Refined			42.40	0.32	0.61	96.60 %
512×512	Baseline	112.47	29.72	50.60	0.32	0.70	94.03 %
	Refined			54.60	0.32	0.71	95.02 %
	GAKN			50.60	0.30	0.64	92.82 %
	GAKN+Refined			54.60	0.31	0.64	94.44 %

TABLE II

COMPARISON OF THE SIZE OF NETWORKS, INFERENCE TIME, POSE ESTIMATION ERRORS AND PERCENT OF ACCEPTED DETECTIONS OF THE NETWORKS WITH DIFFERENT BACKBONES AND DEPTHS OF THE KEYPOINT HEAD.

Heatmap size	Backbone	Convolution layers in the keypoint head	Operations [GFLOPs]	Parameters [M]	Inference time [ms]	Median t _{2D} [m]	Median r _{2D} [deg]	Percent of accepted detections
128×128	HRNet32	1	41.08	28.54 M	35.40	0.46	0.98	93.59 %
	HRNet48		84.10	63.60 M	37.10	0.42	0.72	92.83 %
	HRNet32	3	41.12	28.54 M	35.60	0.43	0.97	93.89 %
	HRNet48		84.18	63.60 M	37.60	0.40	0.71	92.87 %

percentage of accepted detections and has a marginal influence on the number of operations and inference time.

E. Heatmap size

In this subsection, we compare the results of the Baseline network on three different heatmap resolutions. The default implementation of the keypoint detector based on HRNet returns heatmaps which are downsampled four times compared to the input image size, so using an image of 512×512 pixels results in 128×128 pixels heatmaps. The influence on the percentage of accepted detections is marginal, but increasing the heatmap size significantly reduces the translation and rotation error. The difference in the number of operations between a 128×128 and a 256×256 heatmap is small, compared to the 512×512 version, where the number of operations increases three times. This translates into processing time for a single image, wherein the 512×512 version is characterised by the processing time twice as large as in the 128×128 version. Increasing the size of the processed heatmap allows for increased location accuracy, but at the cost of increased processing time. However, there is only a slight increase in the number of network parameters when increasing the heatmap size. This means that all of the architectures discussed, including the largest one, should fit on a graphics card with 8 GB of memory. As the size of the heatmap increased, we achieved a reduction in errors (e.g., the translation error of a 128-sized heatmap relative to a 512-sized heatmap was 26.7%, the rotation error in the same ratio was 33.9%), which was the main research goal. When considering the errors of a 256-sized heatmap, they

were larger than 512 and smaller than a 128-sized heatmap confirming the relationship between heatmap size and location accuracy. Heatmap size does not affect the percentage of accepted detections.

F. Reprojection loss

The configuration of the GAKN network features the same number of operations, parameters, and processing times because the only modification to the baseline network was made during the training phase and does not affect these parameters during inference. There is a reduction in translation and rotation errors in all three heatmap sizes considered. When using reprojection loss for heatmap size 128, there was a 15.2 % reduction in translation error, 37 % in rotation error, for size 256, the translation error decreased by 8.4 %, and rotation error decreased by 15.5 %, whereas for heatmap size 512: there was a 2.6 % reduction in translation error and 9.5 % in rotation error. By combining both approaches - increasing the heatmap size and applying reprojection loss, we obtain the lowest median translation error among all tested models, and a median rotation error comparable to the lowest obtained with heatmap size 256. We believe that using a heatmap size of 256 is a reasonable trade-off between processing time and location accuracy. It achieved the best rotation error result, translation error comparable to the best result, and features a short inference time – similar to the one for 128×128 heatmap. Additionally, the percentage of accepted detections in the GAKN configuration with this heatmap size is higher than in the baseline.

TABLE III
PERCENTAGE OF GROUND-TRUTHS WITHIN UNCERTAINTY ELLIPSE FOR
DIFFERENT STANDARD DEVIATIONS

Standard deviation	1σ	2σ	3σ
Percentage of ground-truths within uncertainty ellipse	48.75 %	87.38 %	98.00 %

G. Reprojection-based refinement

Using reprojection-based refinement does not affect the number of operations performed by the network and the number of parameters, because it is performed at the post-processing stage. On average, this step takes 4 ms per image, which constitutes 11% of the processing time for heatmap 128×128 , and only 8 % for heatmap 512×512 , therefore adding reprojection-based refinement does not increase significantly the image processing time. No clear effect on location accuracy was observed when using reprojection-based refinement. The main goal of this method was to increase the percentage of accepted detections, which was achieved. As a result, we are able to provide a higher number of correct pose estimates, making the localisation system more reliable. By using the GAKN network together with reprojection-based refinement, we eliminate the relatively low percentage of accepted detections.

H. Uncertainty estimation

Using a hand-labeled validation dataset of about 200 images, we evaluated the geometric uncertainty prediction. We checked the percentage of ground truth keypoint locations that are within the 1, 2, and 3 σ uncertainty ellipses. The numerical values are presented in Tab. III.

Qualitative results of covariance matrices prediction are demonstrated in Fig. 5. Comparing Fig. 5A and Fig. 5B, we can notice that the uncertainty of keypoint detection decreases with the distance to the charger. Figure 5C and Fig. 5E show larger uncertainty for poor quality images. On Fig. 5D, the two leftmost points have larger uncertainty along x axis because, from that point of view, estimation of their exact location is ambiguous.

I. Analysis of cumulative distribution plots

Comparing the 3 pairs of graphs (Fig. 6, Fig. 7 and Fig. 8) corresponding to heatmap sizes, it can be seen that the benefit of reprojection loss is biggest for heatmap size 128×128 , and decreases as the heatmap size increases. It can also be seen that by using reprojection-based refinement, we increase the number of accepted detections that provide coarse location information, despite the relatively large translation and rotation error (right part of the graph). Furthermore, we are able to improve the detection accuracy for which the error is below the median (the red line in Fig. 6. is above the green line for values below 0.4 m and 0.6 deg).

J. Network attention analysis

Starting this research we conjectured that the prior geometric knowledge represented by the reprojection-based loss

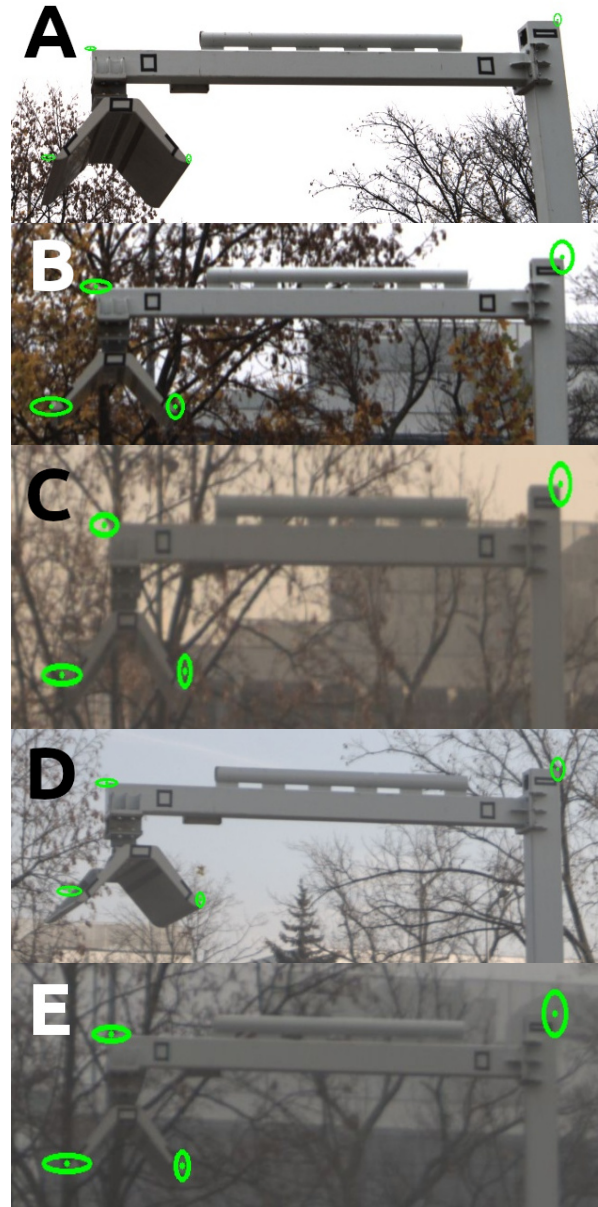


Fig. 5. Visualisation of estimated keypoints locations and 3 sigma uncertainty ellipses for different observation cases. Fig. A presents detection from close distance, B - far distance, C - blurry image, D - different observation angle, E - cloudy/foggy weather

component introduces to the GAKN architecture an inductive bias that helps the network to focus on the most relevant image areas when searching for the keypoints. To verify this claim, we implemented in GAKN an attention analysis layer, based on the Score-CAM method [35]. Compared to the older, gradient-based methods Score-CAM produces results that are visually less noisy, making it easier to interpret the attention depending on the input image.

In Fig. 9 and Fig. 10, the top row (A, B, C and D) shows the activation maps for all 4 detected points in an example image from the test set. Comparing the activation maps shown, it can

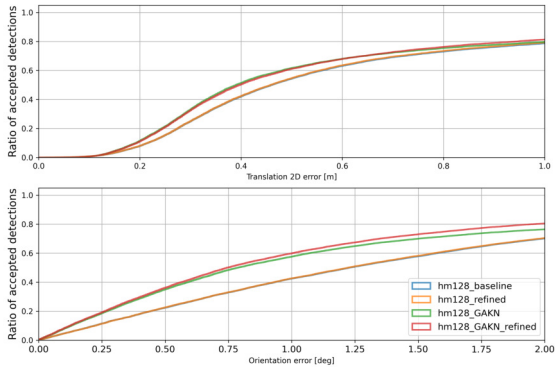


Fig. 6. Cumulative distribution functions of 2D translation error (A) and orientation error (B) for the HRNet32 models with the heatmap size 128×128

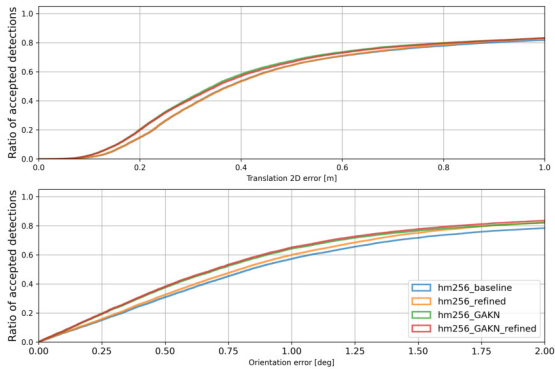


Fig. 7. Cumulative distribution functions of 2D translation error (A) and orientation error (B) for the HRNet32 models with the heatmap size 256×256

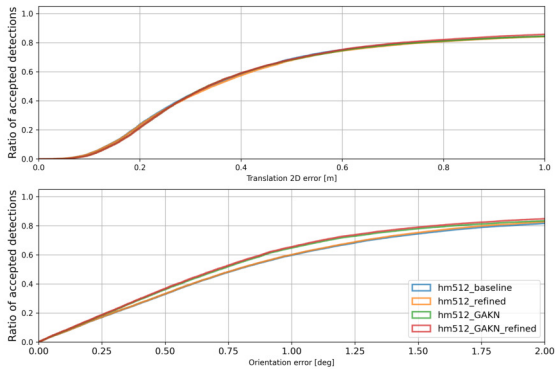


Fig. 8. Cumulative distribution functions of 2D translation error (A) and orientation error (B) for the HRNet32 models with the heatmap size 512×512

be seen that the activations of individual points for the GAKN network are more concentrated and have higher intensity. In the bottom row of Fig. 9 and Fig. 10 (E, F, G, and H), the Score-CAM method marked the parts of the image that had

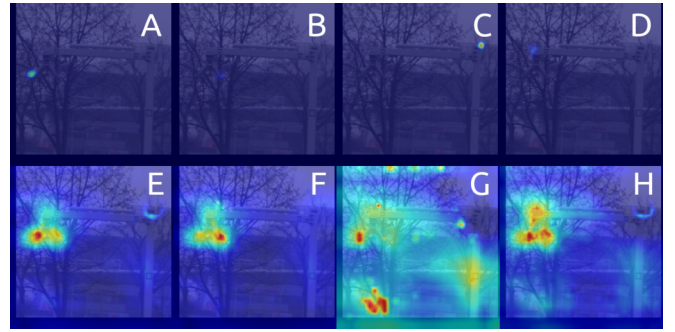


Fig. 9. Heatmaps (top) and the output of the Score-CAM algorithm (bottom) from the baseline HRNet32. Warmer colors mean higher activation

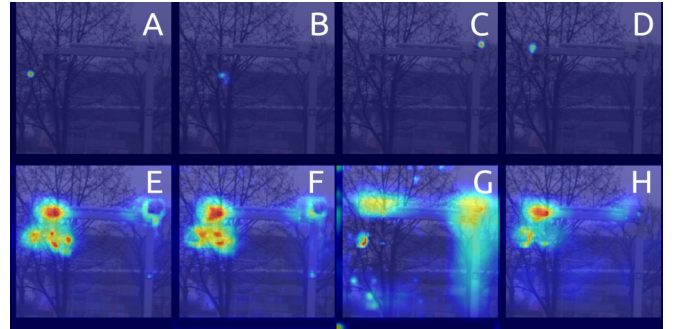


Fig. 10. Heatmaps (top) and the output of the Score-CAM algorithm (bottom) from the Geometric Aware Keypoint Network. Warmer colors mean higher activation

the greatest effect on selecting a specific location in the image as the searched point. The first thing you notice is that in the case of the baseline network, activation is scattered throughout the background image, indicating that the network attention is highly scattered. Furthermore, in the case of point three (G), there is very weak activation near the actual point location and large activation on a portion of the image completely unrelated to the charger structure (lower left corner). In Fig. 10, showing points from the GAKN network, the network's attention is more focused near the searched points than at the baseline.

V. CONCLUSION

This paper evaluated our experience with applying the state-of-the-art HRNet architecture for the detection of keypoints in a challenging scenario of camera pose estimation for localisation of an electric bus. The proposed GAKN model achieved better results than both the Faster R-CNN-based keypoint detector and the baseline HRNet32. Our solution takes less than 50 ms for processing a single image on Nvidia A100, which makes it possible to update the camera pose frequently and facilitates accurate bus localisation along its path to the charging station. Reliable localisation under changing viewpoint, scale, lighting, and weather conditions is achieved without any markers deployed at the charging station. The contributed modifications to the baseline HRNet, based on exploiting the available geometric priors were evaluated positively, as they improve the accuracy of keypoint locations.

Finally, we evaluated the novel elements of GAKN with respect to the localisation accuracy objective, and we have found that using the reprojection loss reduces translation and rotation errors, while using refinement (sanity check) in postprocessing increases the number of accepted detections, thus increasing the availability of the camera pose estimates. Our future research will determine if the proposed GAKN architecture can be scaled down to edge computing platforms, for better cost-efficiency. The project with datasets is available at https://github.com/ZephyrII/mmpose_charger

ACKNOWLEDGMENT


This work is partially under the project “Advanced driver assistance system (ADAS) for precision maneuvers with single-body and articulated urban buses”, co-financed within the Smart Growth Operational Programme 2014-2020 (POIR.04.01.02-00-0081/17-01). P. Skrzypczyński is supported by TAILOR, a project funded by EU Horizon 2020 under GA No. 952215. T. Nowak is supported by PUT internal grant 0214/SBAD/0235.

REFERENCES

- [1] R. I. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge University Press, 2004.
- [2] M. M. Michalek, T. Gawron, M. Nowicki, and P. Skrzypczyński, “Precise docking at charging stations for large-capacity vehicles: An advanced driver-assistance system for drivers of electric urban buses,” *IEEE Vehicular Technology Magazine*, vol. 16, no. 3, pp. 57–65, 2021.
- [3] T. Nowak, M. Nowicki, K. Cwian, and P. Skrzypczyński, “How to improve object detection in a driver assistance system applying explainable deep learning,” in *IEEE Intelligent Vehicles Symposium*, Paris, 2019, pp. 226–231.
- [4] —, “Leveraging object recognition in reliable vehicle localization from monocular images,” in *Automation 2020: Towards Industry of the Future*, ser. AISC, vol. 1140. Cham: Springer, 2020, pp. 195–205.
- [5] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, W. Liu, and B. Xiao, “Deep high-resolution representation learning for visual recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, pp. 3349–3364, 2021.
- [6] M. Toshpulatov, W. Lee, S. Lee, and A. Haghigian Roudsari, “Human pose, hand and mesh estimation using deep learning: a survey,” *The Journal of Supercomputing*, vol. 78, no. 6, pp. 7616–7654, 2022.
- [7] L. G. Clarembaux, J. Pérez, D. Gonzalez, and F. Nashashibi, “Perception and control strategies for autonomous docking for electric freight vehicles,” *Transportation Research Procedia*, vol. 14, pp. 1516–1522, 2016, transport Research Arena TRA2016.
- [8] E. Marchand, F. Spindler, and F. Chaumette, “ViSP for visual servoing: a generic software platform with a wide class of robot control skills,” *IEEE Robotics and Automation Magazine*, pp. 40–52, 2005.
- [9] C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel, and J. D. Tardós, “ORB-SLAM3: an accurate open-source library for visual, visual-inertial, and multimap SLAM,” *IEEE Trans. Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.
- [10] K. L. Lim and T. Bräunl, “A review of visual odometry methods and its applications for autonomous driving,” *arXiv*, vol. 2009.09193, 2020.
- [11] J. Miseikis, M. Ruther, B. Walzel, M. Hirz, and H. Brunner, “3d vision guided robotic charging station for electric and plug-in hybrid vehicles,” *arXiv*, vol. 1703.05381, 2017.
- [12] J. Ma, X. Jiang, A. Fan, J. Jiang, and J. Yan, “Image matching from handcrafted to deep features: A survey,” *International Journal of Computer Vision*, vol. 129, no. 1, pp. 23–79, 2021.
- [13] D. DeTone, T. Malisiewicz, and A. Rabinovich, “Superpoint: Self-supervised interest point detection and description,” in *IEEE Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2018*, 2018, pp. 224–236.
- [14] V. Lepetit, F. Moreno-Noguer, and P. Fua, “EPnP: An accurate o(n) solution to the pnp problem,” *International Journal of Computer Vision*, vol. 81, no. 2, 2008.
- [15] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon, “Bundle adjustment — a modern synthesis,” in *Vision Algorithms: Theory and Practice*. Berlin, Heidelberg: Springer, 2000, pp. 298–372.
- [16] T. Sattler, C. Sweeney, and M. Pollefeys, “On sampling focal length values to solve the absolute pose problem,” in *Computer Vision – ECCV 2014*. Cham: Springer, 2014, pp. 828–843.
- [17] A. Kendall, M. Grimes, and R. Cipolla, “Posenet: A convolutional network for real-time 6-dof camera relocalization,” in *IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 2938–2946.
- [18] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, “PoseCNN: a convolutional neural network for 6D object pose estimation in cluttered scenes,” in *Proceedings of Robotics: Science and Systems*, Pittsburgh, Pennsylvania, June 2018.
- [19] A. Kendall and R. Cipolla, “Geometric loss functions for camera pose regression with deep learning,” in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6555–6564.
- [20] T. Sattler, Q. Zhou, M. Pollefeys, and L. Leal-Taixé, “Understanding the limitations of cnn-based absolute camera pose regression,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 3297–3307.
- [21] B. Zhuang and M. Chandraker, “Fusing the old with the new: Learning relative camera pose with geometry-guided uncertainty,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 32–42.
- [22] Y.-Y. Jau, R. Zhu, H. Su, and M. Chandraker, “Deep keypoint-based camera pose estimation with geometric constraints,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 4950–4957.
- [23] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua, “LIFT: learned invariant feature transform,” in *Computer Vision - ECCV 2016 - 14th European Conference, Proceedings, Part VI*, ser. LNCS, vol. 9910. Springer, 2016, pp. 467–483.
- [24] M. J. Tyszkiewicz, P. Fua, and E. Trulls, “DISK: learning local features with policy gradient,” in *Advances in Neural Information Processing Systems 33 (NeurIPS)*, 2020.
- [25] G. Papandreou, T. Zhu, N. Kanazawa, A. Toshev, J. Tompson, C. Bregler, and K. P. Murphy, “Towards accurate multi-person pose estimation in the wild,” *arXiv*, vol. 1701.01779, 2017.
- [26] J.-J. Liu, Q. Hou, M.-M. Cheng, C. Wang, and J. Feng, “Improving convolutional networks with self-calibrated convolutions,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.
- [27] A. Kendall and R. Cipolla, “Modelling uncertainty in deep learning for camera relocalization,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 4762–4769.
- [28] A. Kendall and Y. Gal, “What uncertainties do we need in bayesian deep learning for computer vision?” *arXiv*, vol. 1703.04977, 2017.
- [29] A. Kumar, T. K. Marks, W. Mou, C. Feng, and X. Liu, “Uglli face alignment: Estimating uncertainty with gaussian log-likelihood loss,” in *IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 2019, pp. 778–782.
- [30] MMPose, “Openmmlab pose estimation toolbox and benchmark,” <https://github.com/open-mmlab/mmpose>, 2020.
- [31] J. Huang, Z. Zhu, and F. Guo, “The devil is in the details: Delving into unbiased data processing for human pose estimation,” *arXiv*, vol. 2008.07139, 2020.
- [32] F. Zhang, X. Zhu, H. Dai, M. Ye, and C. Zhu, “Distribution-aware coordinate representation for human pose estimation,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.
- [33] M. Branch, T. Coleman, and Y. Li, “A subspace, interior, and conjugate gradient method for large-scale bound-constrained minimization problems,” *SIAM J. Sci. Comput.*, vol. 21, pp. 1–23, 1999.
- [34] M. R. Nowicki, “A data-driven and application-aware approach to sensory system calibration in an autonomous vehicle,” *Measurement*, vol. 194, p. 111002, 2022.
- [35] H. Wang, Z. Wang, M. Du, F. Yang, Z. Zhang, S. Ding, P. Mardziel, and X. Hu, “Score-CAM: score-weighted visual explanations for convolutional neural networks,” in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2020.

Analysis of Brain Tumor Using MRI Images

Lewi L. Uberg
 Applied Data Science
 Noroff University College
 Kristiansand, Norway
 lewi@uberg.me 

Seifedine Kadry
 Applied Data Science
 Noroff University College
 Kristiansand, Norway
 seifedine.kadry@noroff.no 

Abstract—The increasing rates of deadly brain tumors in humans correspondingly increase the need for highly experienced medical personnel for diagnosis and treatment. Therefore, to reduce the workload and the time from suspicion of disease to diagnosis, then plan for suitable treatment, there is a need to automate the initial part of the process by implementing a Computer-Aided-Disease-Diagnosis (CADD) system for brain tumor classification. The aim of this research is to develop and deploy all the needed components to design and implement a working Computer-Aided-Disease-Diagnosis (CADD) system for brain tumor classification. First, by understanding the brain tumors themselves and the convention of their classification. Second, the convolution neural network (CNN) structure and functionality and how to manipulate it. Third, find different CNN architectures with promising results in similar tasks. Fourth, find and evaluate the applicability of available data sources. Fifth, implement the most promising solutions and explore the applicability of using transfer learning for the given task to take advantage of previously gained knowledge. Finally, evaluate the results of the experiments, where we show that the DenseNet121 architecture, either fully trained or using transfer learning, likely is the most appropriate candidate for the CADD system in development.

Index Terms—Brain Tumor Classification, Medical Imaging, CADD, Convolutional Neural Networks, Machine Learning, Deep learning.

I. INTRODUCTION

THE CELL of origin and features found when examining the cells tissue define central nervous system tumors and predict their behavior [1]. After meningiomas, the most common primary brain tumor in adults is gliomas, with a rate of 5 to 6 persons per 100,000 annually [2].

The World Health Organization (WHO) tissue classification system categorizes gliomas, with grade 1 as the lowest and grade 4 as the highest. Thus, low-grade gliomas (LGG) consist of grade I and II tumors [3], while high-grade gliomas (HGG) consist of grade III and IV [2]. Grade I are the least malignant or benign tumors. Grade II is relatively slow-growing but may recur as a higher grade. Grade III is malignant and tends to recur as higher grades. Finally, grade IV is the most malignant, aggressive, necrosis and recurrence prone [1].

Histopathologic examination to study tissue morphology, diagnose, and grade brain tumors is the gold standard [3]. However, surgical resection for diagnosing a brain tumor is invasive and risky. Nevertheless, non-invasive diagnostic methods exist, like neuroimaging with Magnetic Resonance Imaging (MRI) [4].

Conventional MRI is the current imaging procedure of choice and identifies tumor size and associated Peritumoral Edema (PTE), one of the main features of malignant glioma [5]. To diagnose the brain tumor without surgical intervention, researchers have developed more advanced imaging methods such as texture analysis, mechanic modeling, and machine learning (ML) that form a predictive multi-parametric image-based model. The application of ML models is an emerging field in radiogenomics and represents a data-driven approach to identifying meaningful patterns and correlations from often complex data sources. ML models train by feeding the ML algorithm a substantial amount of pre-classified image data as input, such as MRI images, to learn which patterns belong to the different classes.

The convolutional neural network (CNN) is a concept introduced by [6] as the Neocognitron, as a model of the brain’s visual cortex; an improvement of [7] previous model for visual pattern recognition. Furthermore, [8] significantly improved the Neocognitron to one of the most successful pattern recognition models, dramatically influencing the field of computer vision and pattern detection in images.

In 2021 a fully automatic hybrid solution for brain tumor classification comprised of several steps was proposed [9]. First, pre-process the brain MRI images by cropping, resizing, and augmenting. Second, use pre-trained CNN models for feature extraction with better generalization. Third, select the top three performing features using fined-tuned ML classifiers and concatenate these features. Finally, use the concatenated feature as input for the ML classifiers to predict the final output for the brain tumor MRI. The proposed scheme uses a novel feature evaluation and selection mechanism, an ensemble of 13 pre-trained CNNs, to extract robust and discriminative features from brain MRI images without human supervision. The CNN ensemble, is comprised of ResNet-50, ResNet-101, DenseNet-121, DenseNet-169, VGG-16, VGG-19, AlexNet, Inception V3, ResNext-50, ResNext-101, ShuffleNet, MobileNet, and MnasNet. Since the researchers use fairly small datasets for training, they take a transfer learning-based approach by using the fixed weights on the bottleneck layers of each CNN model pre-trained on the ImageNet dataset.

II. DATA GATHERING & PRE-PROCESSING

The data-gathering part of the research starts with outlining some criteria. The data should contain non-tumorous, LGG,

and HGG samples and be well balanced between the labels. The latter is essential since too much data augmentation on medical images often does not generalize well. A well-suited candidate was found at The Cancer Imaging Archive (TCIA). The REMBRANDT (REpository for Molecular BRAin Neoplasia DaTa) Dataset [10] seemed to have all this research’s characteristics. The dataset is one of the most trusted publicly available datasets, comprised of MRI scans from 130 subjects of three classes, non-tumorous, LGG, and HGG. Furthermore, the LGG and HGG classes have subclasses that opened the opportunity to make the classification outcome more extensive. The dataset is a combination of metadata from various text and spreadsheet files and the 110,020 images in the DICOM format, which also includes a vast quantity of metadata. After preprocessing, the dataset comprises 123 patients and 105,265 slides, distributed as shown in Table I. At this point, it was discovered that one key bit of information was missing; how to separate the MRI slides that contained the tumorous cells from the ones that did not contain them. All the source data were reexamined, but the answer was not found. While searching online for how to find the needed key, one paper stood out [11]. The paper used that same dataset for a similar application. While being reduced to 4069 slides, it seemed that the authors of this paper had found a way to filter the data further. A meeting with the paper’s main author was arranged to understand how to reproduce the dataset. His research team had employed help from neurologists to go through each slide manually and label them correctly. Fortunately, the main author offered to share a smaller dataset version. After further processing, the dataset has 735 samples distributed, as shown in Table II, and is now ready for training CNN models.

TABLE I
DATASET SAMPLE DISTRIBUTION

Disease	Grade	Label	Unique Samples	Count
Astrocytoma	II	LGG	30	25286
Astrocytoma	III	HGG	17	16038
GBM	IV	HGG	43	32837
Non-Tumorous	n/a	n/a	15	17041
Oligodendroglioma	II	LGG	11	9335
Oligodendroglioma	III	HGG	7	4728
Total			123	105265

TABLE II
MANUALLY LABELED DATASET SAMPLE DISTRIBUTION

Label	Sample Count
Normal	168
LGG	287
HGG	280
Total	735

III. METHODOLOGY

Finding the CNN architecture with or without transfer learning that yields the best results for the classification task is the primary goal of the current research. Therefore designing a

pipeline where the CNN architecture can easily be substituted is essential. Doing so required finding the generally most suitable division of the Test, Validation, and Train subsets, the most suitable baseline hyperparameter settings, and whether image augmentation was to be used. For this task, VGG16 [12] was selected, as it had been used for the same application with promising results [13] [14]. After training a large number of VGG16 models, a baseline for training other CNN architectures was established. A total of 60 images, 20 from the three classes, were randomly picked for the Test set. The remaining images are divided into 30% and 70% for the Validation and Training sets. The Training set is also augmented with rotation, width and height shift, shearing, zoom and brightness adjustment, and horizontal flipping. Filters with minimal adjustments to the original gave better results than fewer with more extensive adjustments. The Normal label has a little over half of the samples as LGG and HGG, and is therefore producing a higher number of augmented images. The Normal label in the Train set is increased from 103 to 309, LGG from 186 to 372, and HGG from 182 to 364. With a batch size of 24, the baseline uses 10 epochs for training with a learning rate of 0.001, Adam for optimization with categorical cross-entropy as the loss function. In addition to the VGG16 architecture, five other CNN architectures were used to train models, including AlexNet, MobileNet [15], ResNext [16], DenseNet121 [17], and a custom-designed architecture. The custom CNN architecture is designed to accept N amount of 224x224 image matrices with 3 color channels; it consists of three convolutional layers of 32, 64, and 64 filters, of filter size 3x3, with zero-padding, and ReLU as the activation function. The first convolutional layer is followed by a max-pooling layer of pool size 4x4, and the last two convolutional layers have a pool size of 2x2, each followed by a dropout layer with a dropout rate of 0.15. Next, a flattening layer is added to transform matrix output to vector inputs to be accepted by the first dense layer, which is comprised of 512 ReLU activated neurons, followed by a dropout layer with a 0.5 dropout rate. The last hidden layer is a dense layer of 256 ReLU activated neurons. The model’s final layer, its output, is a 3 neuron softmax activated dense layer for classification. The general architecture of this model is shown in Fig. 1, and the flowchart is provided in Fig. 2.

IV. RESULTS

All the CNN architectures implemented to this point show outstanding results, as shown in Table III. While multiple runs were made on each architecture to obtain the given results, and there is an indication that the Test set in particular needs expansion, the results indicate that DenseNet121 is the best candidate for the given classification task. Surprisingly enough, with the custom architecture as the runner-up.

V. RESEARCH CONTRIBUTIONS

The research conducted has been a good start for further developing a usable CADD system. A pipeline for data preprocessing that makes preparing new data samples both easy and

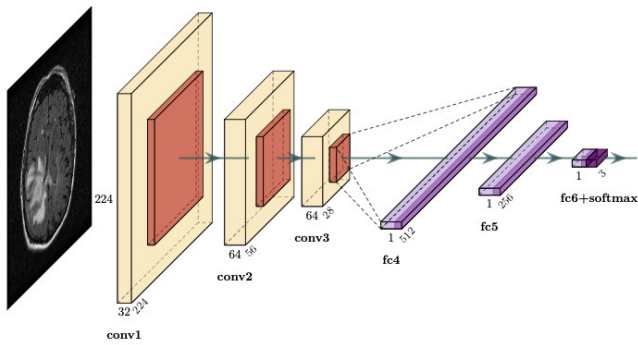


Fig. 1. Custom CNN Architecture.

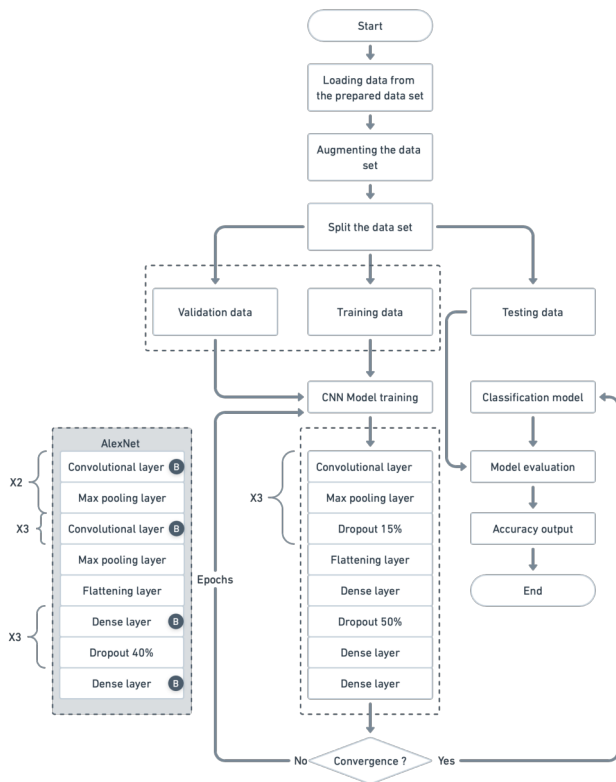


Fig. 2. Flowchart of the custom framework.

TABLE III
CNN ARCHITECTURE RESULTS

Architecture	Epochs	Initial learning rate	Loss	Accuracy & F1
VGG16	10	0.001	0.3218	0.933
ResNext	12	0.001	0.1244	0.933
MobileNet	28	0.001	0.1141	0.950
Custom	22	0.000316	0.0989	1.00
AlexNet	28	0.001	0.0582	0.983
DenseNet121	12	0.001	0.0254	1.00

fast has been developed. Similarly, a pipeline for augmenting the training data and a pipeline for defining CNNs, training them, and providing a classification model as the output has been developed successfully. Also, to demonstrate the potential of the research, a user interface has been developed and deployed to a web server where the general public can accessed at tumorclass.info. All the source code used in this project can be found in the [GitHub repository](#).

VI. DEPLOYMENT

A service called ngrok was selected for the deployment. Ngrok is a free service that allows the exposure of localhost to the internet in a secure manner. With ngrok installed, the only steps needed were to run the FastAPI in uvicorn on the local machine and then run ngrok in a separate terminal to expose the local host to the internet. To make it a little more elegant, a domain name was purchased, and the DNS was set to give access to ngrok. A monthly subscription on ngrok was also purchased to connect the autogenerated ngrok domain name to the custom domain name. In figures Fig. 3, Fig. 4, Fig. 5, and Fig. 6 the implemented user interface is shown.

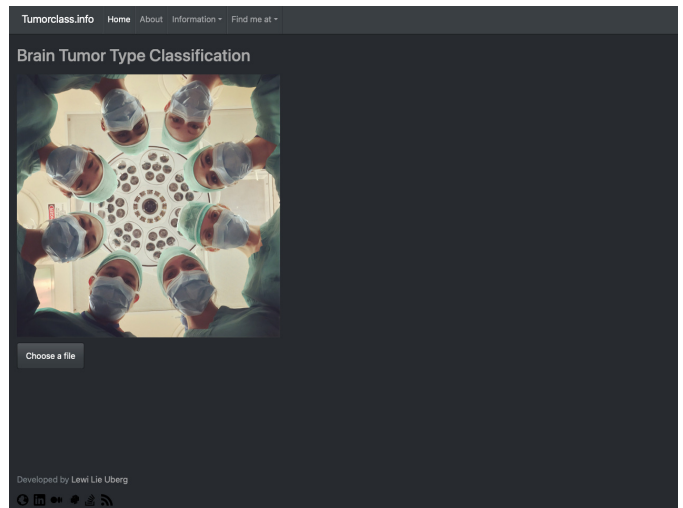


Fig. 3. Interface of our platform.

VII. CONCLUSION & FUTURE WORK

The next step after our CADD system is to expand the data set. The amount of suitable data available is limited. However, the REMBRANDT dataset has a vast number of images suitable for manual labeling performed by radiologists. Expanding the data set will make it more transparent, which architectures generalize well. As such, these architectures can be the focus of further development. These days, MRI scans provide 3D images, so further development to facilitate 3D classification is also a viable option.

REFERENCES

- [1] D. N. Louis, H. Ohgaki, O. D. Wiestler, W. K. Cavenee, P. C. Burger, A. Jouvett, B. W. Scheithauer, and P. Kleihues, "The 2007 WHO classification of tumours of the central nervous system," *Acta neuropathologica*, vol. 114, no. 2, pp. 97,109, Aug. 2007.

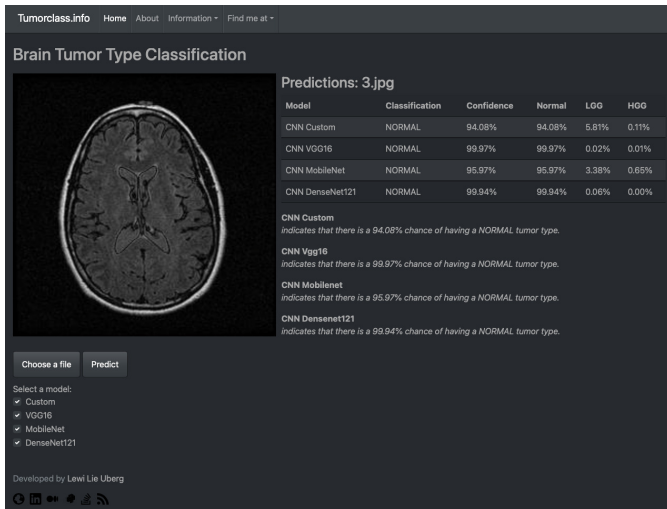


Fig. 4. Classification demo for normal MRI.

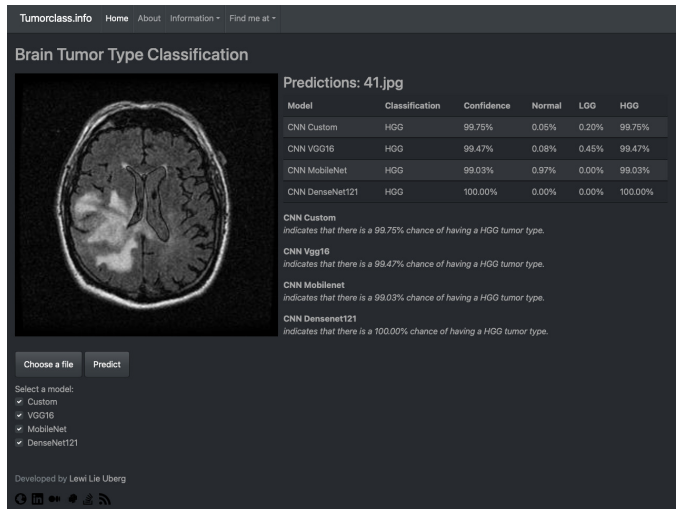


Fig. 6. Classification demo for HGG MRI.

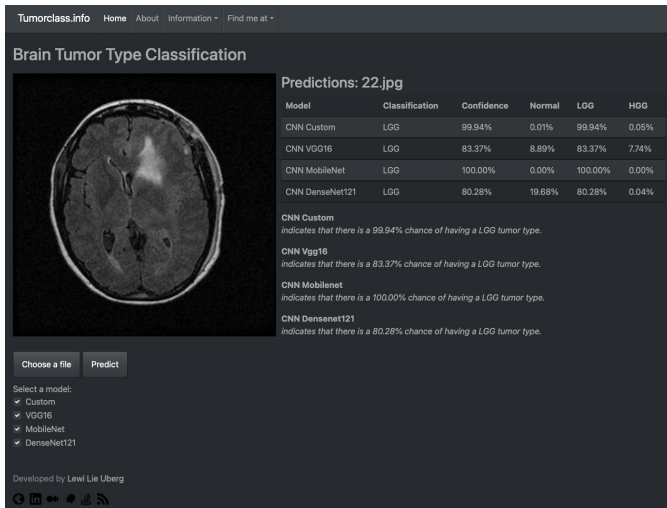


Fig. 5. Classification demo for LGG MRI.

- [2] L. S. Hu, A. Hawkins-Daarud, L. Wang, J. Li, and K. R. Swanson, "Imaging of intratumoral heterogeneity in high-grade glioma," *Cancer letters*, vol. 477, pp. 97,106, May 2020. [Online]. Available: <https://pubmed.ncbi.nlm.nih.gov/32112907>
- [3] D. A. Forst, B. V. Nahed, J. S. Loeffler, and T. T. Batchelor, "Low-grade gliomas," *The oncologist*, vol. 19, no. 4, pp. 403,413, Apr. 2014. [Online]. Available: <https://pubmed.ncbi.nlm.nih.gov/24664484>
- [4] Q. Luo, Y. Li, L. Luo, and W. Diao, "Comparisons of the accuracy of radiation diagnostic modalities in brain tumor: A nonrandomized, nonexperimental, cross-sectional trial," *Medicine*, vol. 97, no. 31, Aug. 2018. [Online]. Available: <https://pubmed.ncbi.nlm.nih.gov/30075495>
- [5] C.-X. Wu, G.-S. Lin, Z.-X. Lin, J.-D. Zhang, L. Chen, S.-Y. Liu, W.-L. Tang, X.-X. Qiu, and C.-F. Zhou., "Peritumoral edema on magnetic resonance imaging predicts a poor clinical outcome in malignant glioma," *Oncology Letters*, vol. 10, no. 5, pp. 2769,2776, Aug. 2015. [Online]. Available: <https://www.spandidos-publications.com/10.3892/ol.2015.3639>
- [6] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biological Cybernetics*, vol. 36, no. 4, pp. 193,202, Apr. 1980. [Online]. Available: <https://link.springer.com/article/10.1007/BF00344251>
- [7] —, "Cognitron: A self-organizing multilayered neural network,"

- Biological Cybernetics*, vol. 20, no. 3, pp. 121–136, Sep. 1975. [Online]. Available: <https://doi.org/10.1007/BF00342633>
- [8] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278,2324, Nov. 1998. [Online]. Available: <http://yann.lecun.com/exdb/publis/pdf/lecun-98.pdf>
- [9] J. Kang, Z. Ullah, and J. Gwak, "MRI-Based Brain Tumor Classification Using Ensemble of Deep Features and Machine Learning Classifiers," *Sensors*, vol. 21, no. 6, Mar. 2021. [Online]. Available: <https://www.mdpi.com/1424-8220/21/6/2222>
- [10] L. Scarpace, A. E. Flanders, R. Jain, T. Mikkelsen, and D. W. Andrews, "Data From REMBRANDT [Data set]," 2019. [Online]. Available: <https://wiki.cancerimagingarchive.net/display/Public/REMBRANDT#35392299515cc672b974080a1394cbe9c649c74>
- [11] S. Khawaldeh, U. Pervaiz, A. Rafiq, and R. S. Alkhalwaldeh, "Noninvasive Grading of Glioma Tumor Using Magnetic Resonance Imaging with Convolutional Neural Networks," *Applied Sciences*, vol. 8, no. 1, 2018. [Online]. Available: <https://www.mdpi.com/2076-3417/8/1/27>
- [12] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," in *International Conference on Learning Representations*, Sep. 2015. [Online]. Available: <https://arxiv.org/pdf/1409.1556.pdf>
- [13] O. N. Belaid and M. Loudini, "Classification of Brain Tumor by Combination of Pre-Trained VGG16 CNN," *Journal of Information Technology Management*, vol. 12, no. 2, pp. 13,25, 2020. [Online]. Available: https://jitm.ut.ac.ir/article_75788.html
- [14] O. Sevli, "Performance Comparison of Different Pre-Trained Deep Learning Models in Classifying Brain MRI Images / Beyin MR Görüntülerini Sınıflandırmada Farklı Önceden Eğitilmiş Derin Öğrenme Modellerinin Performans Karşılaştırması," *Acta Infologica*, vol. 5, p. 2021, Jun. 2021. [Online]. Available: <https://cdn.istanbul.edu.tr/file/JTA6CLJ8T5/99DD9C496BF14E44859851B33E49A006>
- [15] A. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," *ArXiv*, 04 2017. [Online]. Available: <https://arxiv.org/pdf/1704.04861.pdf>
- [16] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated Residual Transformations for Deep Neural Networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Jul. 2017, pp. 5987,5995. [Online]. Available: <https://ieeexplore.ieee.org/document/8100117>
- [17] G. Huang, Z. Liu, G. Pleiss, L. Van Der Maaten, and K. Weinberger, "Convolutional Networks with Dense Connectivity," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1,1, 2019. [Online]. Available: <https://ieeexplore.ieee.org/document/8721151>

Insights into Neural Architectures for Learning Numerical Concepts from Simple Visual Data

Andrzej Śluzek

Warsaw University of Life Sciences-SGGW
 ul. Nowoursynowska 166, 02-787 Warszawa, Poland
 Email: andrzej_sluzek@sggw.edu.pl

Abstract—The paper reports some results on neural architectures for learning numerical concepts from visual data. We use datasets of small images with single-pixel dots (*one to six per image*) to learn the abstraction of small integers, and other numerical concepts (e.g. *even versus odd numbers*). Both fully-connected and convolutional architectures are investigated. The obtained results indicate that two categories of numerical properties apparently exist (in the context of discussed problems). In the first category, the properties can be learned without acquiring the counting skills, e.g. the notion of *small, medium and large numbers*. In the second category, explicit learning of counting is embedded into the architecture so that the concepts are learned from numbers rather than directly from visual data. In general, we find that CNN architectures (if properly crafted) are more efficient in the discussed problems and (additionally) come with more plausible explainability.

I. INTRODUCTION AND MOTIVATION

THE CONCEPTS of numbers and numerical properties develop primarily (e.g. [1], [2]) from sensory experiences, with visual inspection playing the pivotal role.

Researchers investigate the topic of learning numerical abstractions mainly (apart from actual and prospective applications) as a challenging AI/ML problem.

Initially, the works were focused on object counting rather than understanding the abstractions, with efforts on *counting-by-localizing* sub-tasks, e.g. [3]. This was an application-oriented approach, and some sources (e.g. [4]) indicate that true understanding of numbers may not be even needed to perform counting tasks in visual data.

Later, researchers expressed more interests in grasping/learning the concept of numbers and numerical abstractions. First, it was done in the context of human brain functioning (e.g. [5]) but recent works focus on machine learning aspects as well. In particular, visual data have been explicitly used as inputs to learning algorithms and architectures, e.g. [6], [7], [8]. Complexity of those visual inputs is usually limited to avoid complicated image processing sub-tasks, i.e. either binary [6], [7] or near-binary [8] low-resolution images are used.

In this paper, we follow the same approach. Using results of [6] and [8] as the starting point, we attempt to develop simple neural architectures for learning the concept of numbers (from a limited range 1 to 6) and other numerical abstractions which can exist within such a narrow range of integers. Examples of such abstract concepts are:

- *even* and *odd* numbers;
- *small* (1, 2), *medium* (3, 4) and *large* (5, 6) numbers;
- etc.

Formally, each concept is a division of integers (from 1 to 6 range) into classes. For example, enumeration from 1 to 6 corresponds to six classes (1), (2), (3), (4), (5) and (6), *even* and *odd* numbers form two classes (1, 3, 5) and (2, 4, 6), etc. Then, an input image should be assigned to the correct class based on the number of dots it contains.

In Section II, we explain the proposed methodology and overview the developed datasets. The considered neural architectures are also briefly explained. Section III presents the conducted experiments and achieved performances. Informal explanations of the trained architectures are provided there as well. The concluding Section IV highlights the most significant facts, underlines unsolved problems and proposes directions for the future work.

II. METHODOLOGY

A. Assumptions

In [6], two neural models are proposed for estimating numbers of white non-overlapping rectangles (up to 10) in small black images of 28×28 resolution.

In [7], more general (but still simple) tasks are discussed, i.e. estimating either the numbers of white isolated pixels or white connected components in 256×256 black images. The assumed numbers of counted objects are much larger (e.g. up to 3,000 isolated pixels).

In [8], the task is to count isolated pixels (up to 10) in 10×10 images. However, the images are only approximately binary with random polarity (bright pixels on dark backgrounds or another way around).

In this paper, the proposed scenario (based on the above approaches) is further simplified. Images are very small (7×7) and contain up to 6 dots. Such small numbers of dots are motivated by a well known psychological fact that humans can visually perceive (without explicit counting) at most 7 objects. The images are only approximately binary and their polarity is random.

We consider both fully-connected (FcNN) and convolutional (CNN) neural networks. This is motivated by inconclusive reports from the past papers, where *pros* and *cons* of both

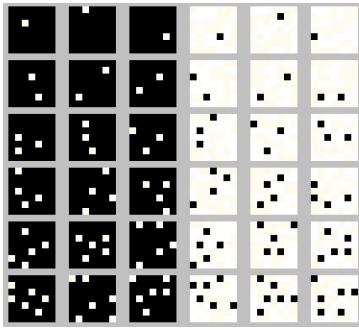


Fig. 1. Exemplary dataset images.

TABLE I
FIRST CNN ARCHITECTURE (CNN1).

Layer	Parameters	Activation
input	7×7	
conv.	$16 \times 3 \times 3$, str=1	relu
maxpool	2×2 , str=1	
conv.	$16 \times 16 \times 2 \times 2$, str=1	relu
fc	10 outputs	tanh
fc	6 (or 2, or 3) outputs	softmax
output	6 (or 2, or 3)	

architectures are highlighted. Obviously, the considered architectures are rather simple to reflect small size and low complexity of input images.

B. Datasets

The developed dataset consists of 12,000 7×7 near-binary images (6,000 dark images with bright dots and 6,000 images of the opposite polarity). The numbers of dots range from 1 to 6 (each number in 2,000 images).

Near-binary images are used to alleviate overfitting, and to more realistically represent the real-world visual conditions.

Figure 1 shows a small sample of dataset images.

For the actual training and testing, the dataset is divided into two parts of 6,000 images each (with the same ratio of all types of images). Only one half is used for training and validation, while the other half is used for testing only.

C. Neural architectures

As mentioned earlier, both FcNN and CNN architectures are considered. After extensive *try-and-error* tests (and partially following the ideas from [6] and [8]) we eventually propose two CNN and two FcNN architectures shown in Tables I- IV.

It can be noticed that (CNN1, CNN2) and (FcNN1, FcNN2) pairs are very similar. Actually, 'variant 2' architectures are obtained by embedding 'variant 1' (with 6 nodes in the terminal layer) and adding an additional FC layer. In the embedded 'variant 1', its last layer is assumed to learn numbers from 1 to 6. Thus, the terminal layer of 'variant 2' architectures would infer the numerical abstractions from presumably learned concepts of enumeration. This is further explained in Section III (with some special cases separately discussed).

TABLE II
SECOND CNN ARCHITECTURE (CNN2).

Layer	Parameters	Activation
input	7×7	
conv.	$16 \times 3 \times 3$, str=1	relu
maxpool	2×2 , str=1	
conv.	$16 \times 16 \times 2 \times 2$, str=1	relu
fc	10 outputs	tanh
fc	6 outputs	softmax
fc	2 (or 3) outputs	softmax
output	2 (or 3)	

TABLE III
FIRST FCNN ARCHITECTURE (FcNN1)

Layer	Parameters	Activation
input	49	
fc	53 outputs	tanh
fc	10 outputs	tanh
fc	6 (or 2, or 3) outputs	softmax
output	6 (or 2, or 3)	

Actually, the architectures can be much slimmer than presented here. For example, the first hidden layer of FcNN architectures can be reduced from 53 nodes to, for example, 20 nodes. The current size is proposed to satisfy theoretical conditions of approximation theorems.

Similarly, the number of filters in convolutional layers of CNN architectures can be much smaller (see Section III); the proposed numbers just provide safe margins.

III. EXPERIMENTS AND RESULTS

The proposed architectures were separately trained for various concepts, including:

- learning the concept of numbers from 1 to 6;
- differentiating between *even* and *odd* numbers;
- differentiating between *medium* (3, 4) numbers and other numbers (i.e. either *small* (1, 2) or *large* (5, 6));

TABLE IV
SECOND FCNN ARCHITECTURE (FcNN2)

Layer	Parameters	Activation
input	49	
fc	53 outputs	tanh
fc	10 outputs	tanh
fc	6 outputs	softmax
fc	2 (or 3) outputs	softmax
output	2 (or 3)	

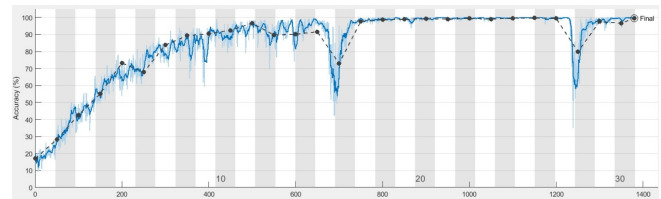


Fig. 2. Training (continuous line) and validation (circles) accuracy plots for CNN1 learning numbers from 1 to 6.

		Target class					
		1000	16	0	0	0	0
Output class	1000	1000	16	0	0	0	0
	0	0	974	0	0	0	0
	0	0	10	996	0	0	0
	0	0	0	4	998	2	0
	0	0	0	0	2	984	2
	0	0	0	0	0	14	998
		Target class					
		1000	1000	0	0	0	0
		0	0	1000	0	0	0
		0	0	0	1000	3	0
		0	0	0	0	997	8
		0	0	0	0	0	991
0	0	0	0	0	1	993	

Fig. 3. Confusion matrices for FcNN1 (top) and for CNN1 (bottom) trained to identify numbers from 1 to 6.



Fig. 4. Examples of images with incorrectly identified numbers of dots by CNN1.

- and a few other (sometimes slightly artificial) numerical concepts.

In the following subsections, we discuss how the architectures can handle the above tasks.

A. Learning numbers from 1 to 6

This task is the only one similar to the problems discussed in earlier papers (e.g. [4], [6], [7], [8], [9]).

We found that both CNN1 and FcNN1 architectures can easily learn to nearly flawlessly recognize the number of dots in test dataset images. As an example, Fig. 2 shows training performances of CNN1.

The overall accuracies (on the test dataset, see Section II-B) are almost the same, i.e. 99.68% for CNN1 and 99.17% for FcNN1. However, the confusion matrices (given in Fig. 3) indicate more plausible performances of CNN1. Errors are located within classes with larger numbers of dots, i.e. in the scenarios where humans can make mistakes more frequently.

Examples of some incorrectly identified images are given in Fig. 4. At the first glance, they may look confusing even for humans.

B. Learning even and odd numbers

In this task, classification is not directly related to magnitudes of numbers but to their specific quantifiers (which should be identified by trained networks). In [5], it is argued that in human perception knowledge of numbers makes important contributions to acquiring meanings of such quantifiers. Our observations confirm this conclusion.

We find that 2-output architectures of CNN1 and FcNN1 type (i.e. the hidden layer for learning numbers is missing) are apparently unable to learn the abstraction of *even* and *odd*

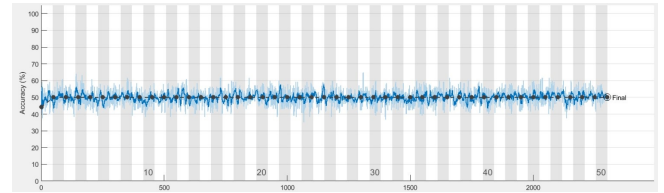


Fig. 5. Training (continuous line) and validation (circles) accuracy plots for CNN1 learning *even* and *odd* numbers.

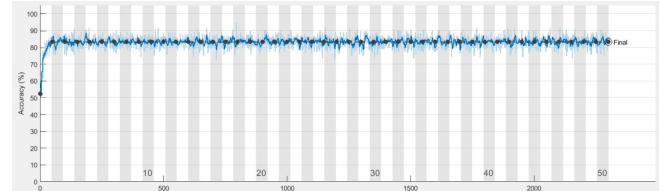


Fig. 6. Training (continuous line) and validation (circles) accuracy plots for CNN2 learning *even* and *odd* numbers.

numbers. The most typical accuracy for CNN1 is the random-choice 50.00%. FcNN1 performs slightly better, reaching 60.48%. Thus, the architectures are unable to learn such numerical abstractions.

An exemplary plot of CNN1 performances during training is shown in Fig. 5.

When 'variant 2' architectures are used, the situation changes. Although accuracy of FcNN2 architecture deteriorates compared to FcNN1 (58.52% versus 60.48%), a significant improvement can be noticed for CNN2. Accuracy reaches 83.32% and training is very fast, as shown in Fig. 6.

In other words, if the concept of numerical values is embedded in the process (as explained in Section II-C) the convolutional architecture is able to quickly generalize the abstraction of *even* and *odd* numbers with reasonable accuracy. This can be considered a kind-of-projection of [5] observations onto machine learning domain.

C. Learning other numerical abstractions

The other exemplary numerical abstractions considered in the experiments are:

- *medium* numbers, i.e. (3, 4) versus all other numbers,
- various classes of *compact* and *disconnected* subsets of integers (see Table V).

In the first experiment, the overall conclusions are similar to Section III-B. CNN1 architecture achieves results equivalent to random choice (66.67% accuracy; a training plot shown in Fig. 7) while FcNN can reach 74.36%.

Again, a switch from CNN1 to CNN2 architecture significantly improves performances. With the concept of numbers embedded, CNN2 achieves almost perfect 99.38% accuracy, and the learning curve is very steep (see Fig. 8). For FcNN2, however, there is no real improvement.

In other experiments, various classes have been proposed within the range 1, 6; examples are shown in Table V. We

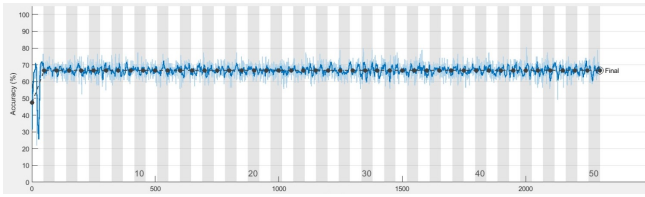


Fig. 7. Training (continuous line) and validation (circles) accuracy plots for CNN1 learning to distinguish $\{3, 4\}$ from other numbers.

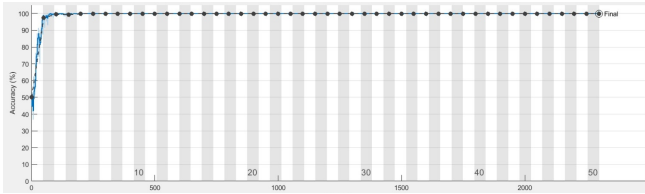


Fig. 8. Training (continuous line) and validation (circles) accuracy plots for CNN2 learning to distinguish $\{3, 4\}$ from other numbers.

found that learning abilities of proposed architectures strongly depend on whether the classes consist of compact subsets of integers (e.g. *even* and *odd* numbers are disconnected classes).

For FcNN architectures (regardless the variant) the results are satisfactory only if the classes are *compact*. Table V contains a number of examples with the top accuracies obtained by FcNN architectures (either FcNN1 or FcNN2). The accuracy is always very high for *compact* classes, while for *disconnected* classes the results are unacceptable (often even below the level of random choice).

For CNN architectures the results are generally much better. For *compact* classes, almost perfect accuracies can be obtained both by CNN1 and CNN2, while for *disconnected* classes only CNN2 are able to reach very high accuracies. CNN1 architectures usually struggle to go beyond the random-choice level.

However, some interesting examples of *disconnected* classes have been identified, where even CNN1 are able to score near-perfect performances. Learning numbers divisible by 3 (i.e. $\{3, 6\}$ and $\{1, 2, 4, 5\}$ classes) is one of such examples.

As expected, FcNN architectures perform rather poorly (63.25% for FcNN1 and 64.92% for FcNN2) in this example. However, CNN architectures can almost perfectly grasp this abstraction, even in CNN1 variant (i.e. without the embedded concept of numericals). Accuracy reaches 99.97% for the test

TABLE V
ACCURACY OF FCNN ARCHITECTURES IN SELECTED OTHER PROBLEMS.

Classes	accuracy	compact?
$(1, 2)(3, 4)(5, 6)$	99.1%	YES
$(1)(2)(3, 4, 5, 6)$	99.5%	YES
$(1, 4)(2, 5)(3, 6)$	56.1%	NO
$(2, 4)(1, 3, 5, 6)$	74.7%	NO
$(3, 5)(1, 2, 4, 6)$	68.3%	NO
$(4, 5)(1, 2, 5, 6)$	74.1%	NO

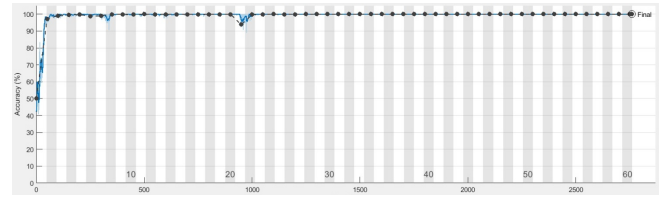


Fig. 9. Training (continuous line) and validation (circles) accuracy plots for CNN1 learning numbers divisible by 3.

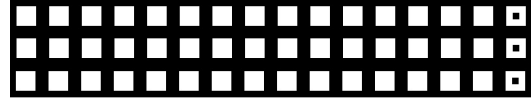


Fig. 10. First-layer of 3×3 filters for (from top to bottom): CNN1 for learning 1 to 6 numbers, CNN2 for learning *even* and *odd* numbers and CNN1 for learning numbers divisible by 3.

dataset, and the learning curve is very steep (see Fig. 9).

Very similar results are obtained for another (somehow artificial) case, where the first class consists of $(2, 6)$ and all other numbers form the second class.

D. Explainability issues

The complexity of proposed architectures is (deliberately) very low to correspond to low complexity of the input data. Therefore, it is possible to inspect the learned values of NN parameters, and informally explain their roles in the process.

Again, the conclusions are very different for FcNN and CNN architectures. In FcNN structures, it is hardly possible to identify any intuitive explanations for the obtained weights. The numbers look random, and even if some specific features (e.g. near-zero columns in the weight matrix of the second hidden layer) can be noticed, they do not exist in other nets trained to learn similar concepts.

For CNN structures, however, the convolutional layers of successfully trained architectures are almost identical. Fig. 10 shows visual representations of 3×3 filters of the first convolutional layer for several such cases. It looks obvious that in all of them only one filter plays an active role (others are approx. averaging filters) and its apparent role is to detect isolated pixels.

Fig. 11 presents the corresponding filters for an unsuccessfully trained CNN1 (see Fig. 5) for learning *even* and *odd* numbers. Again, only one active filter can be noticed, but its effective functionality is unclear. Thus, the failure of this architecture can be attributed to unsuccessful building of the first-layer filters.

Similarly, 2×2 filters of the second convolutional layer are virtually identical for all CNN architectures, and (again) only



Fig. 11. First-layer of 3×3 filters in CNN1 unsuccessfully trained to distinguish between *even* and *odd* numbers.

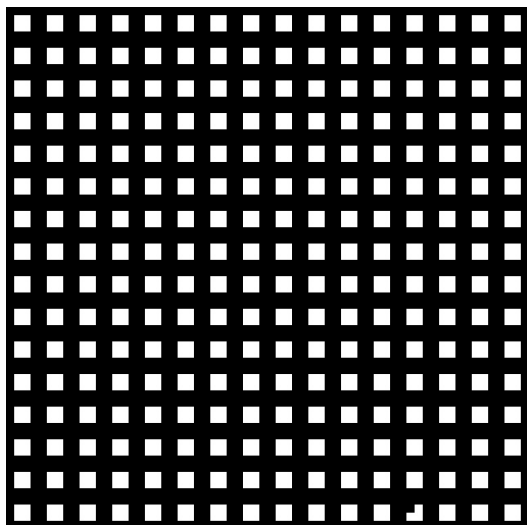


Fig. 12. Second-layer of 2×2 filters for CNN architectures. They are virtually the same for all trained CNN1 and CNN2 structures. Only one filter (in the last row) seems to perform actively.

one of the filters seems active (see Fig. 12). Apparently, its role is to accumulate large-magnitude instances of the first-layer results, i.e. to effectively count the number of dots.

Thus, the inferred explanations accurately correspond to the common-sense understanding of the investigated learning tasks.

IV. SUMMARY

The paper presents some insights into processes of learning basic numerical concepts from visual data (small near-binary images containing several single-pixel dots) by using simple FcNN and CNN architectures.

It was found that in case of concepts with classes forming *compact* subsets in the domain of integers (e.g. (1), (2) and (3, 4, 5, 6)) both FcNN's and CNN's can be trained to achieve nearly perfect accuracy on test datasets. However, for *distributed* classes (e.g. *even* and *odd* numbers) the conclusions are less straightforward.

For *distributed* classes, FcNN architectures are sometimes able to achieve accuracies above the random-choice level, but generally their performances are unsatisfactory. Typically, CNN's in CNN1 variant also do not go above the random-choice levels. However, CNN2 architectures (where learning numbers is embedded into a hidden layer of the net) can achieve very high (or almost perfect) accuracies even for *distributed* classes. There are, nevertheless, specific cases where both CNN2 and CNN1 architectures can achieve almost perfect accuracy after very short learning period (with very steep learning curve). Further experiments and analysis are needed to better understand this phenomenon.

In many aspects, the obtained results supplement conclusions from past works on similar topics (mainly [6], [7] and [8]). In particular, we can (partially) support opinions

from [7] about limited usefulness of fully connected architectures for counting tasks in visual data. The concepts of using small-scale CNN's for learning numerical abstractions (proposed in [6]) is also confirmed.

However, conclusions from our earlier work [8] about limited capabilities of CNN architectures should be significantly revised.

Additionally, we found that in the investigated problems CNN architectures can be much better explained. In particular, the weights of convolutional filters nicely correspond to the intuitive understanding of the problems. In FcNN architectures, we did not find any regularities coinciding with the nature of learned problems.

Last but not least, our investigations can be (distantly) related to works discussing learning numerical concepts from neuropsychological perspectives. For example, relations between numerical and logical quantifiers are discussed in [5], while early stages of sensory-based counting abilities and understanding numbers by animals (from insects to humans) are presented in [1], [10], [11], [12] (and many other works). We believe that similar investigations can be continued in the domain of AI agents and systems.

REFERENCES

- [1] A. J. Kersey and J. F. Cantlon, "Primitive concepts of number and the developing human brain," *Language Learning and Development*, vol. 13, no. 2, pp. 191–214, 2017. doi: 10.1080/15475441.2016.1264878
- [2] M. H. Fischer and S. Shaki, "Number concepts: abstract and embodied," *Phil. Trans. Royal Society B*, vol. 373, no. 1752, p. 20170125, 2018. doi: 10.1098/rstb.2017.0125
- [3] E. Walach and L. Wolf, "Learning to count with cnn boosting," in *Proceedings of the 14th European Conference on Computer Vision, part II*, vol. LNCS 9906, 2016. doi: 10.1007/978-3-319-46475-6_41 pp. 660–676.
- [4] S. Sabathiel, J. L. McClelland, and T. Solstad, "Emerging representations for counting in a neural network agent interacting with a multimodal environment," vol. ALIFE 2020: The 2020 Conference on Artificial Life, 2020. doi: 10.1162/isal_a_00333 pp. 736–743.
- [5] V. Troiani, J. E. Peelle, R. Clark, and M. Grossman, "Is it logical to count on quantifiers? dissociable neural networks underlying numerical and logical quantifiers," *Neuropsychologia*, vol. 47, no. 1, pp. 104–111, 2009. doi: <https://doi.org/10.1016/j.neuropsychologia.2008.08.015>
- [6] C. Creatore, S. Sabathiel, and T. Solstad, "Learning exact enumeration and approximate estimation in deep neural network models," *Cognition*, vol. 215, p. 104815, 2021. doi: 10.1016/j.cognition.2021.104815
- [7] S. Guan and M. Loew, "Understanding the ability of deep neural networks to count connected components in images," in *2020 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*, 2020. doi: 10.1109/AIPR50011.2020.9425331 pp. 1–7.
- [8] A. Śluzek, "Counting dots: On learning numerical concepts from visual data," *Proceedings of the 3rd Polish Conference on Artificial Intelligence, April 2022, Gdynia, Poland*, pp. 16–19, 2022.
- [9] M. Fang, Z. Zhou, S. Chen, and J. L. McClelland, "Can a recurrent neural network learn to count things?" *Cognitive Science*, 2018.
- [10] A. Cope, E. Vasilaki, D. Minors, C. Sabo, J. Marshall, and A. Barron, "Abstract concept learning in a simple neural network inspired by the insect brain," *PLoS Computational Biology*, vol. 14, no. 9, p. e1006435, 2018. doi: 10.1371/journal.pcbi.1006435
- [11] M. Tomonaga and T. Matsuzawa, "Enumeration of briefly presented items by the chimpanzee (*pan troglodytes*) and humans (*homo sapiens*)," *Animal Learning & Behavior*, vol. 30, p. 143–157, 2002. doi: 10.3758/BF03192916
- [12] K. Wynn, "Children's understanding of counting," *Cognition*, vol. 36, no. 2, pp. 155–193, 1990. doi: 10.1016/0010-0277(90)90003-3

1st Workshop on Personalization and Recommender Systems

RECOMMENDER Systems are present in our everyday life while we reading news, logging in to social media or buying something at e-shops. Thus, it is not surprising that this domain is getting more and more attention from researchers from academia as well as from industry practitioners. However, the way in which they look at the same problem differs a lot.

Personalization is an important element in novel recommendation techniques. Nonetheless, it is a wider topic that concerns also user modelling and representation, personalized systems, adaptive educational systems or intelligent user interfaces.

The objective of PeRS is to extend the state-of-the-art in Personalization and Recommender Systems by providing a platform at which industry practitioners and academic researchers can meet and learn from each other. We are interested in high quality submissions from both industry and academia on all topics related to Personalization and Recommender Systems.

TOPICS

The list of topics includes, but is not limited to:

- Personalization
 - User Profiles
 - Ontology-based user models
 - Personalized systems
 - Intelligent user interfaces
- Recommender Systems approaches
 - Collaborative Recommender Systems
 - Semantic-based Recommender Systems
 - Context-aware Recommender Systems
 - Cross-domain Recommender Systems
- Machine Learning techniques for Recommender Systems
 - Association Rules
 - Clustering methods
 - Neural Networks
 - Deep Learning
 - Reinforcement Learning
- Applications of Recommender Systems methods
 - News recommendations
 - Tourism recommendations
 - Fashion recommendations
 - Podcasts recommendations
 - Medical recommendations
 - Other domain-specific recommenders
- Evaluation of Recommender Systems

- Metrics
- Evaluation studies
- Reproducibility of existing methods
- Case studies of real-world implementations

TECHNICAL SESSION CHAIRS

- **Karpus, Aleksandra**, Gdańsk University of Technology, Poland
- **Przybyłek, Adam**, Gdańsk University of Technology, Poland

PROGRAM COMMITTEE

- **Anelli, Vito Walter**, Politecnico University of Bari, Italy
- **Aziz Butt, Shariq**, University of Lahore, Pakistan
- **Borg, Markus**, SICS Swedish ICT AB, Sweden
- **Brzeski, Adam**, Gdańsk University of Technology, Poland
- **Cellary, Wojciech**, WSB Universities in Poznan, Poland
- **Dedabrishvili, Mariam**, International Black Sea University, Georgia
- **de Gemmis, Marco**, University of Bari "Aldo Moro", Italy
- **Dutta, Arpita**, National University of Singapore, Singapore
- **Ghofrani, Javad**, University of Lübeck, Germany
- **Goczyla, Krzysztof**, Gdańsk University of Technology, Poland
- **Inayat, Irum**, National University of Computer and Emerging Sciences, Pakistan
- **Lops, Pasquale**, University of Bari "Aldo Moro", Italy
- **Madeyski, Lech**, Wrocław University of Technology, Poland
- **Marcinkowski, Bartosz**, University of Gdańsk, Poland
- **Misra, Sanjay**, Ostfold University College, Halden, Norway
- **Mohapatra, Durga Prasad**, NIT Rourkela, India
- **Mukta, Saddam Hossain**, United International University, Bangladesh
- **Ng, Yen Ying**, Nicolaus Copernicus University, Poland
- **Nguyen, Phuong T.**, University of L'Aquila, Italy
- **Nocera, Francesco**, Politecnico University of Bari, Italy
- **di Noia, Tommaso**, Politecnico University of Bari, Italy
- **Orłowski, Cezary**, WSB University in Gdańsk, Poland
- **Polignano, Marco**, University of Bari "Aldo Moro", Italy
- **Poniszewska-Maranda, Aneta**, Lodz University of Technology, Poland

- **Szymański, Julian**, Gdańsk University of Technology, Poland
- **Taweel, Adel**, Birzeit University, Palestine
- **Theobald, Sven**, Fraunhofer IESE, Germany
- **Tkalcic, Marko**, University of Primorska, Slovenia
- **Vagliano, Iacopo**, Amsterdam University Medical Center, Netherlands
- **Wrycza, Stanisław**, University of Gdańsk, Poland

User Experience and Multimodal Usability for Navigation Systems

Lumbardha Hasimi

Institute of Information Technology,
Lodz University of Technology, Lodz, Poland
ORCID: 0000-0003-3001-350X
Email: lumbardha.hasimi@dokt.p.lodz.pl

Aneta Poniszewska-Marańda

Institute of Information Technology,
Lodz University of Technology, Lodz, Poland
ORCID: 0000-0001-7596-0813
Email: aneta.poniszewska-maranda@p.lodz.pl

Abstract—User experience as a concept of human-computer interaction is crucial for the evaluation of systems and applications. Every new generation of navigational systems provides new features and extended functionality, which has additional functions that can oftentimes confuse the primary information on the system’s functionality. The conducted experiment analyses and observes available versions of navigation systems such as Garmin Drive 50 and/or TomTom Go, which are offered with certain advantages on features. The paper presents the selected aspects regarding the implementation, design, environment, recruitment, tests, and evaluation of the navigation systems, concluding from the results that there is difference between audio and visual mode.

Index Terms—navigation systems, human-computer interaction methods, user experience, multimodal usability

I. INTRODUCTION

New generation of navigational systems provide new features and extended functionality, which can sometimes be source of confusion on the primary information of the system’s functionality. This often times lead to poor usability, especially when it comes to operations of the primary functions [1], [5]. In order to design a proper environment with effect on user satisfaction, the usability aspect should be taken into account. This is to ensure successful performance on the primary operations of the system, as well as on the upgraded versions, regardless of the users’ experience and ability with it. User experience as a concept of human-computer interaction is a fundamental concept in the evaluation of systems and applications, particularly because it affects issues such as usability, cognitive load, affective experiences, mental demand, efficiency, etc.

The evaluation of usability is quite diverse. As an area, it is under uninterrupted and active development, meaning new approaches and procedures come as continuous expansion of the practice into several contexts [2]. Hence, the testing methodologies can be different based on aim, place, and moderating style. However, in general there are two evaluation methods on usability, namely qualitative and quantitative [2], [6]. The quantitative evaluation is required to generally get numerical values that represent the level of usability, whilst qualitative evaluation is required to identify and examine issues and problems. In our work, we focus mainly on the

quantitative method as a primary evaluation, namely surveys designed for specific tasks.

The concept of cognitive load, in the other hand, referring to the amount of effort required while performing an action, is an aspect taken into consideration mainly because of potential interference with the processes in the use of a system [17]. Considering that an ideal application or system ensures an interface that keeps user cognitive load to a minimum, what we have considered as an important indicator [18].

Additionally, mental workload, considered as a demand placed on the user by the system, depends on many parameters, which lead to the fact that the result of a task performed is subjective to the users’ experience and the ability [7]. For this reason, it was determined that during the recruitment process the experience, initial capacities, reaction time and fatigue will be taken into account. The conducted experiment analyses and observes available navigation systems such as Garmin Drive 50 and/or TomTom Go, which are available and with certain advantages on features, although not as widely used as the most traditional available Maps. The compactible devices used were Camper/RV 890 and/or TomTom GO 60S Automotive GPS Navigation Device. The main reasons for carrying out this experiment are to find out the satisfaction and efficiency of the applications used and how much of the aimed aspects are satisfactory. This experiment is carried out aiming to analyse and compare user experience on aspects as efficiency, usability, and cognitive load on system’s multimodal mode.

The defined null hypothesis assumes that there is slightly no difference between visual or audio mode of feedback in the user experience. Hence, it was decided on three defined dependent variables, namely efficiency, usability, and cognitive load, where the efficiency regards how fast the user finished the task, taking into account its completion time. Furthermore, usability as the second variable is defined as how easy the feedback is learned by the user. Finally, the mental and/or physical load for the tasks being taken by the user and the difference between audio and visual feedback. The independent variable is the feedback type with the given conditions on the audio and visual feedback treatments, shown in table I. The environment of the design is certainly basic, as it contains only one independent variable with two possible conditions based on the mode of the feedback. The design would be affected

by different user features, such as background, accessibility, previous experience which affects efficiency and other minor variables [5], [14], [15].

Aspects regarding the implementation, design, environment, recruitment, tests and evaluation are thoroughly discussed in the next sections. The paper is structured as follows: section 2 presents the recruitment process, selected profile and confounding variables together with the methods, tasks and setup. Section 3 deals with the result, statistics and significance of each test performed. Finally, section 4 gives an overall conclusion and discussion on final findings.

II. METHODOLOGY OF THE STUDY

The recruitment design is selected considering different factors, concentrating especially on the depended variable – the user experience, which should consider that has a wide range of variations. In the view of certain complex tasks that require information retrieval from the system, problem-solving skill of a certain level is necessary for multimodal feedback. Certainly, some individual users might have limitations, disabilities, or accessibility issues, hence within group design would be best to eliminate such differences [3], [4], [16].

Participant's profile is selected deciding in various factors in order to minimize the confounding factors:

- Users with similar experience in using such systems.
- User of similar age group.
- User with/without IT background.
- Accessibility or level of disability (visual or audio issues/impairments).

Training and overview of the proposed navigation system is given to the user and an orientation session conducted to make participants familiar with the tasks required and how to tackle it. Moderators are not allowed to interfere during any ongoing activity, only in case of error issues, to allow completion of the activity.

In this setting the dependent variable is User Experience for the following categories, i.e., Efficiency, Usability and Cognitive Load. Given it with only one independent variable which is the type of feedback for the chosen systems, taking into consideration two conditions/states: Audio and Visual feedback for the respective system.

Whereas the environment of the carried experiment consisted of several elements. The first session was conducted to assess the navigation system usability, first by being introduced to the use of the system. Secondly, specific tasks for a specific period of time and path. It was followed by a session with the test environment where participants were asked to explore the virtual landscape freely and navigate from one place to another. The participants were instructed to try all of the navigation functions provided.

A short questionnaire was handed out after the road/path completion and the participants answered the questions within d minutes. The moderator followed a set of predefined questions during the later session. Preparation required before the study starts includes the consent form, demographic question-

naire, participants coding and preparing all other questionnaires.

The tasks described in table I shall be assigned to the user walking or riding a car. This was not given, and in order to avoid any confounding variables, it could be provided.

Conditions intended are audio and visual feedback. The order of the condition to which the participant was assigned is the key to avoiding other factors or confounding variables. To avoid variables or other factors on the evaluation of dependent variable, it is very important to use randomization in order to lessen the effects. Such variables can be fatigue, or the learning curve while trying the tasks of audio than visual mode in order.

The conditions are counterbalanced among 20 participants using a simple randomization technique. The certain sequencing considering only two options of sequencing: $C1\beta C2$ or $C2\beta C1$, having in mind the counterbalance of the order effect, for $n = 20$, gives times 10 for each sequence, hence it was assigned which participant gets what sequence in order, having randomly "1" or "0" assigned and counted equal, followed by code of sequences.

After conducting the test, every participant filled in two questionnaires, one for the SUS (System Usability Scale) and second for TLX (Task Load Index). Using the adopted version of the questionnaire for system usability scale to measure the user experience for both audio and visual feedback [6], [7] as presented in the tables II and III. Similarly, some questions were adopted from user satisfaction questionnaire GUESS [9].

The questions defined in table III are required as a post questionnaire for participants who completed the tasks that had been discussed to measure the usability. For the efficiency measurement [8], the evaluation was done taking into consideration two features: the task completion time and the speed of conducting it, taken into units; time in seconds and speed per minute respectively. After the measurement of efficiency for both modes of the system, using T-Test was possible to compare the two means of speed and task completion for audio and visual feedback.

III. SYSTEM USABILITY SCALE (SUS) TESTS OF NAVIGATION SYSTEMS

Usability studies are well suited for gathering qualitative or quantitative behavioural data and for answering design-related questions. Considering that our setup is rather focused on collecting feedback on the features rather than attitudinal feedback [10], [11], [12], it is well fit for our purpose. Nevertheless, data collected through methods mentioned, was later on described and analysed with Paired Sample T-Test. System usability score was calculated based on the questionnaires filled by participants [13].

Table IV contains the results from preliminary questionnaire, calculated and elaborated accordingly to the System Usability Scale (SUS) test score. In a general view the Visual mode tends to draw higher values than the Audio one (Fig. 1). Similarly, as seen in the results from boxplot the mean SUS scores of visual values are higher than the audio values.

TABLE I
TASKS AND SETUP.

Scheduled routes	Schedule a route in advance, either by given destination, decided route according to the path and time given	≈ 45min
Specific destination	Type in an address from a location outside of the city center (e.g., Suburban area in proximity to the city) you want to travel and follow a direction on the navigation system using the feedback type assigned.	≈ 90min
Find a nearby medical center	Find closest hospital/clinic specialized in surgery and follow feedback from a navigation system	≈ 45min
Completion	Wrap-up, archiving logs and documents, collection of results.	≈ 45min

TABLE II
SYSTEM USABILITY SCALE QUESTIONS.

Sentence	Score (1-5)
1. I think that I would like to use this system frequently.	Score (1-5)
2. I found the system unnecessarily complex.	Score (1-5)
3. I thought the system was easy to use.	Score (1-5)
4. I think that I would need the support of a technical person to be able to use this system.	Score (1-5)
5. I found the various functions in this system were well integrated.	Score (1-5)
6. I thought there was too much inconsistency in this system.	Score (1-5)
7. I would imagine that most people would learn to use this system very quickly.	Score (1-5)
8. I found the system very cumbersome to use.	Score (1-5)
9. I felt very confident using the system.	Score (1-5)
10. I needed to learn a lot of things before I could get going with this system.	Score (1-5)

TABLE III
TASK LOAD INDEX QUESTIONNAIRE TO ASSESS THE WORKLOAD OF TASKS FOR GIVEN CASES.

Subject	Question	Scale elaboration
Mental Demand (MD)	How mentally demanding was the task?	Very low to Very high
Physical Demand (PD)	How physically demanding was the task?	Very low to Very high
Temporal Demand (TD)	How hurried or rushed was the pace of the task?	Very low to Very high
Performance (P)	How successful were you in accomplishing what you were asked to do?	Failure to Perfect
Effort (E)	How hard did you have to work to accomplish your level of performance?	Very low to Very high
Frustration (F)	How insecure, discouraged, irritated, stressed, and annoyed were you?	Very low to Very high

TABLE IV
CALCULATED SUS SCORES FOR VISUAL AND AUDIO FEEDBACK.

PID	Visual mode	Audio mode
1	55	27.5
2	80	37.5
3	30	12.5
4	70	100
5	85	32.5
6	72.5	85
7	95	100
8	50	45
9	97.5	50
10	80	95
11	67.5	42.5
12	100	100
13	57.5	7.5
14	62.5	82.5
15	82.5	45
16	57.5	22.5
17	42.5	30
18	95	95
19	40	37.5
20	82.5	45

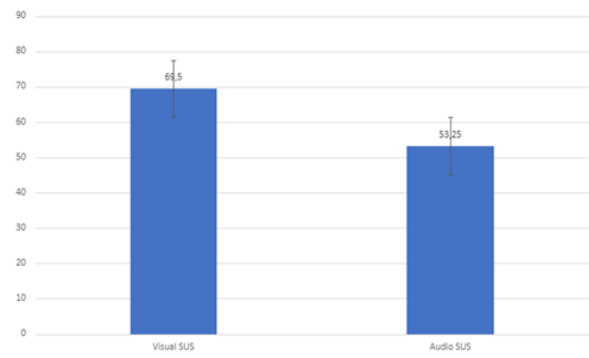


Fig. 1. Boxplot showing mean SUS scores of visual and audio values.

Measure 1	Measure 2	t	df	p
Visual SUS	- Audio SUS	2.778	19	0.012

Fig. 2. Paired sample t-test.

Considering that only statistics are not enough to show the significance in the system usability, paired samples T-Test shall be applied. Furthermore, Standard Error of Mean (SEM) and Standard Deviation (SD) are higher for audio over the visual

feedback for which the values in audio are not very close to mean values, compared to ones in visual feedback (Table V).

Conducted Paired Samples T-Test on calculated SUS scores resulted in significant effect, $t(19) = 2.778, p = 0.012$

TABLE V
DESCRIPTIVE STATISTIC FOR THE DUAL MODE.

Descriptive Statistics	AUDIO	VISUAL
System Usability Scale		
Valid	20	20
Missing	0	0
Mean	53.25	69.50
Standard Error of Mean (SEM)	7.30	4.51
Standard Deviation (SD)	32.66	20.19
Variance	1067.17	407.63
Minimum	7.50	30.00
Maximum	100.00	100.00

TABLE VI
DESCRIPTION STATISTICS FOR VISUAL AND AUDIO FEEDBACK.

	N	Mean	Standard Deviation	Standard Error
MDVIS	20	6.857	4.214	0.796
MDAUD	20	9.250	5.803	1.097
PDVIS	20	7.893	4.653	0.879
PDAUD	20	6.964	4.647	0.878
TDVIS	20	4.821	3.389	0.640
TDAUD	20	5.857	4.461	0.843
PVIS	20	8.750	4.436	0.838
PAUD	20	9.000	4.431	0.837
EVIS	20	8.500	5.037	0.952
EAUD	20	9.679	4.877	0.922
FVIS	20	5.750	4.024	0.761
FAUD	20	9.821	5.976	1.129

(Fig. 2). Results of Paired Samples T-Test conclude that the differences between is significant which means that the population (SUS scores for different feedback systems) have intrinsic differences. The p-value of 0.012 is much smaller than 0.05, so it is possible to reject the null hypothesis of no difference between type of feedback in system usability and consider with a high degree of confidence 98.8% that visual type of feedback is giving better usability. It is shown in table VI, that Fatigue and Mental Demand have significance, results have p value smaller than 0.05, so we can say high confidence that there is difference in Mental Demand and Fatigue factor effected user experience for both Audio and Visual feedback.

The observed result were $t(27) = -3.170$ and $p = 0.004$ for the Mental Demand, whilst for the Fatigue $t(27) = -3.663$ and $p = 0.001$. So null hypothesis that there is no difference in user experience between audio and visual feedback is rejected. Due to the significance test results applied.

IV. CONCLUSIONS

The main goal of this paper was to find out whether there is a difference in user experience and system usability, in case of audio and visual feedback of navigation systems. The paper further investigated how using audio or visual mode impacts the variables user experience, usability and cognitive load in given setup. The results of the experiment show that there is a considerable difference in user experience for the System Usability and task cognitive load for the two types of audio and visual feedback for navigation systems, as shown in the results of the tests engaged. As for system usability dependent variable Visual Feedback was preferred among all participants

with 98.8% confidence, whereas the degree of freedom showed the result: $t(19) = 2.778, p = 0.012$. On the other hand, for the Task Cognitive Load as dependent variable, the results showed that only two factors have significant effect in its Task Load, namely Fatigue and Mental Demand where p resulted smaller than 0.05, what leads to confidence of 99% that Audio type feedback requires more mental demand and causes more Tiredness to the users. These data resulted on given scores: $t(27) = -3.170, p = 0.004$ for Mental Demand and $t(27) = -3.663, p = 0.001$ for Fatigue. Consequently, the results lead to the rejection of null hypothesis that there is no difference in user experience between audio and visual feedback.

REFERENCES

- [1] P. Zalewski and B. Muczynski, Extended Framework for Usability Testing in e-Navigation Systems, *TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation*, Vol. 10, pp. 43–48, 2016.
- [2] G. Papatzani, Evaluating usability evaluation methods for location-aware interactive systems in contextually rich environment, <https://qmro.qmul.ac.uk/xmlui/handle/123456789/1275>, 2011.
- [3] R.L. Charles and J. Nixon, Measuring mental workload using physiological measures: A systematic review, *Applied Ergonomics*, Vol. 74, pp. 221–232, 2019.
- [4] C. Marchand, J.B. De Graaf and N. Jarrasse, Measuring mental workload in assistive wearable devices: a review, *J NeuroEngineering Rehabil*, Vol 18(10), pp. 160, 2021.
- [5] C.M. Barnum, Preparing for usability testing, in *Usability Testing Essentials, Ed. Morgan Kaufmann, 2nd edition*, 2021.
- [6] P. Laubheimer, Beyond the NPS: Measuring Perceived Usability with the SUS, NASA-TLX, and the Single Ease Question After Tasks and Usability Tests, Nielsen Norman Group, <https://www.nngroup.com/articles/measuring-perceived-usability/>, 2018.
- [7] A.J. Lazard, J.S.B. Brennen and S.P. Belina, App Designs and Interactive Features to Increase mHealth Adoption: User Expectation Survey and Experiment, *JMIR Mhealth Uhealth*, Vol. 9(11), pp. e29815, 2021.
- [8] R. Minelli, A. Mocci and M. Lanza, Measuring Navigation Efficiency in the IDE, *7th International Workshop on Empirical Software Engineering in Practice (IWESEP)*, pp. 1–6, 2016.
- [9] M.H. Phan, J.R. Keebler and B.S. Chaparro, The Development and Validation of the Game User Experience Satisfaction Scale (GUESS), *Human Factors*, Vol. 58(8), pp. 1217–1247, 2021.
- [10] J. Sauro, Measuring Usability with the System Usability Scale (SUS), *MeasuringU*, 2011.
- [11] S. Radmard, A.J. Moon and E.A. Croft, Interface design and usability analysis for a robotic telepresence platform, *Proceedings of 24th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pp. 511–516, 2015.
- [12] M. Schrepp, A. Hinderks and J. Thomaschewski, User Experience mit Fragebögen evaluieren – Tipps und Tricks für Datenerhebung, Auswertung und Präsentation der Ergebnisse, *UP 2016, Gesellschaft für Informatik e.V. und die German UPA e.V.*, 2016.
- [13] K. Tcha-Tokey, O. Christmann, E. Loup-Escande and S. Richir, Proposition and Validation of a Questionnaire to Measure the User Experience in Immersive Virtual Environments, *IJVR*, Vol. 16(1), 2016.
- [14] B. Borowska, Learning Competitive Swarm Optimization, *Entropy*, Vol. 24(2), 283, MDPI, ISSN 1099-4300, pp. 1–17, 2022.
- [15] T. Galaj, F. Pietrusiak, M. Galewski, R. Ledzion and A. Wojciechowski, Hybrid Integration Method for Sunlight Atmospheric Scattering, *IEEE Access*, Vol. 9, Publisher IEEE, ISSN 2169-3536, pp. 40681–40694, 2021.
- [16] S. Zakrzewski, B. Stasiak, T. Klepaczka and A. Wojciechowski, VR-oriented EEG signal classification of motor imagery tasks, *Human Technology*, Vol 18 (1), ISSN 1795-6889, pp. 29–44, 2022.
- [17] F. Paas, J. E. Tuovinen, H. Tabbers and P. W. M. Van Gerven, Cognitive Load Measurement as a Means to Advance Cognitive Load Theory, *Educational Psychologist*, vol. 38(1), pp. 63–71, Jan. 2003.
- [18] K. Whitenon, Minimize Cognitive Load to Maximize Usability, Nielsen Norman Group. <https://www.nngroup.com/articles/minimize-cognitive-load/> (accessed Jul. 14, 2022).

Utilizing Frequent Pattern Mining for Solving Cold-Start Problem in Recommender Systems

Eyad Kannout*, Michał Grodzki[†] and Marek Grzegorowski[‡]

Institute of Informatics, University of Warsaw

Banacha 2, Warsaw, Poland

Email: *eyad.kannout@mimuw.edu.pl, [†]m.grodzki@students.mimuw.edu.pl, [‡]m.grzegorowski@mimuw.edu.pl

Abstract—Although several approaches have been proposed throughout the last decade to build recommender systems (RS), most of them suffer from the cold-start problem. This problem occurs when a new item hits the system or a new user signs up. It is generally recognized that the ability to handle cold users and items is one of the key success factors of any new recommender algorithm. This paper introduces a frequent pattern mining framework for recommender systems (FPRS) - a novel approach to address this challenging task. FPRS is a hybrid RS that incorporates collaborative and content-based recommendation algorithms and employs a frequent pattern (FP) growth algorithm. The article proposes several strategies to combine the generated frequent itemsets with content-based methods to mitigate the cold-start problem for both new users and new items. The performed empirical evaluation confirmed its usefulness. Furthermore, the developed solution can be easily combined with any other approach to build a recommender system and can be further extended to make up a complete and standalone RS.

Index Terms—recommendation system, cold-start problem, frequent pattern mining, quality of recommendations.

I. INTRODUCTION

OVER the past few decades, alongside the explosion in the amount of data on the internet, the popularity of online streaming services, e-commerce, and social media has highlighted an important challenge to provide users with recommendations that match their preferences and interests. Therefore, the demand for finding more efficient techniques to generate recommendations has received more attention. Over the years, researchers have suggested various approaches for building recommender systems that leverage the rating history and possibly some other information, such as users' demographics and items' characteristics. The majority of these approaches can be classified into three main categories: (i) collaborative filtering (CF), (ii) content-based (CB) filtering, and (iii) hybrid filtering.

The basic idea behind collaborative filtering is that users with similar tastes or preferences tend to behave similarly in the future. This technique relies on historical transactions to compute similarities among users from which the recommendations are eventually generated. An analogous approach can be applied to create recommendations based on item similarities. On the other hand, content-based filtering tries to utilize items' characteristics, users' demographics, and

contextual information to recommend additional items similar to those preferred by the target user in the past. Finally, hybrid techniques cover the weaknesses and exploit the strengths of CF and CB models by combining them to provide more relevant results.

The aforementioned techniques are highly appreciated by practitioners and businesses. However, they also encounter significant difficulties in terms of data characteristics. One of the issues is related to the sparsity of data. The discussed methods rely on modeling the user-item interactions, and hence, the quality of such may be impacted by an insufficient number of movies rated by each user. Another challenge is related to the so-called cold-start problem. This phenomenon is particularly inconvenient and occurs whenever recommendations are generated for a new item or user that does not have any interaction or rating in the history. In fact, many state-of-the-art recommendation algorithms may generate unreliable recommendations for such cases since they cannot learn the preference embedding of these new users/items [1], [2], [3].

In this study, we presented a particular take on the challenge of devising more effective and efficient recommendation techniques. We put special attention to properly handling the new users and items. We propose several methods to overcome the cold-start problem and the sparsity nature of the datasets by utilizing the FP-growth algorithm to generate frequent patterns based on items' characteristics and users' demographics. The main contributions of this paper are as follows:

- 1) We introduce *frequent pattern mining framework for recommender systems* (FPRS) - a novel hybrid recommender system that utilizes the FP-growth algorithm to produce frequent itemsets based on the ratings in the user-item matrix.
- 2) We utilize the items' and users' features to extract particular patterns based on the features selected.
- 3) We propose several techniques to mitigate the cold-start problem by using the discovered patterns to provide recommendations for new users and involve new items into the recommendations generated for the users.
- 4) We conduct the empirical evaluation of the proposed approach on well established benchmark data (MovieLens 100K and MovieLens 1M), showing its effectiveness in the presence of new entities (users and items, i.e., movies) in test data.

Research co-funded by Polish National Science Centre (NCN) grant no. 2018/31/N/ST6/00610.

The remainder of this paper is organized as follows. Section II describes and reviews important research efforts that addressing the cold-start problem in the domain of Recommender systems. In Section III, we provide background information for collaborative filtering and frequent pattern mining. In Section IV, we present a novel frequent pattern mining model (FPRS) that utilizes the ratings in user-item rating matrix to discover the frequent itemsets associated with selected users/items features. Section V evaluates and compares the proposed model with a baseline recommender system. Finally, in Section VI, we draw conclusions and suggest possible future work.

II. RELATED WORKS

Recommender systems (RS) predict the utility of an item to a user and suggest the best items concerning the user's preferences, where the items may represent movies, books, restaurants, or any other things [4], [5]. The aforementioned capability of RSs makes those techniques especially useful, and indeed, there are many areas of their successful applications like eCommerce, online marketing, or social networks [6], [7]. The scientific literature provides several taxonomies for RS [8]. However, the most common approaches refer to content-based or collaboration-based techniques and their various hybridizations [9]. Collaborative Filtering (CF) is one of the most widely used and successful techniques, with excellent results in a wide range of applications in many fields [8], hence is particularly interesting in our research and further reviewed in detail in Section III-A. Despite the noticeable decline in their popularity in favor of collaborative systems, content-based techniques are still widely used because of handling the so-called cold-start problem [10]. Because of the significantly different characteristics of those approaches, it is advisable to construct hybridizations of both [11], as further discussed in our study.

A typical RS consists of the three main elements: a user model (established by analyzing the users' interests and preferences), an item model (based on its characteristics), and the recommendation algorithm that is a key constituent. There are many reported approaches to implementing the recommendation algorithm by the specific adoption of machine learning (ML) models like matrix factorization, deep neural networks, or factorization machines (FM) [12], [13], [14]. Building RS on top of the state-of-the-art ML models leveraged the quality of recommendation results, improving user satisfaction and profits in e-commerce [15], [6], [16]. At the same time, however, we may observe the known problems with ML related to the data sparsity, the latency of prediction returned by complex models, and foremost, the unfairness of recommendations for new users or items that is often referred to as the *cold-start* problem [17], [18].

Solving scalability issues is one of the most common tasks when deploying big-scale recommender systems. Especially as the number of users and items significantly grows over time, it is essential for RSs to handle requests without appreciable

latency. This problem is particularly challenging for memory-based methods like k-nearest neighbors. However, in the case of web-scale recommendation tasks like social media, the Internet of Things (IoT), or various e-commerce applications, it is a hot topic also for model-based techniques, especially considering more complex and deep models [1], [19]. Another aspect that is particularly noticeable for collaborative filtering is related to the sparsity of user-item interactions [20]. Here, the quality of CF-based methods may be impacted by an insufficient number of items rated by each user [21]. Some recommender systems suffer from their over-specialization (sometimes referred to as a serendipity problem). It is observed when the RS produces recommendations with minimal novelty, i.e., all of the same kind [22]. Recently, there is also an increasing interest in privacy awareness when handling user data and explainability of recommendations [23], [24].

Regardless of recent achievements in RS, the cold-start problem is still one of the most prevailing topics deserving further attention and is particularly interesting in the context of our study [3], [21]. The difficulty arises due to the deficient information about new entities. Therefore it has a particularly strong negative impact on collaborative methods, heavily impacting the fairness of recommendations for new users, often passing over new items [18]. Most of the attempts to deal with such a problem consider enhancing the collaborative-based methods with content-based approaches that leverage the intrinsic characteristics of the analyzed entities. For example, in [2], the authors propose hybrid recommender models that use content-based filtering and latent Dirichlet allocation (LDA)-based models. Whereas in [9], we may find a hybrid RS that combines the singular-value decomposition-based collaborative filtering with content-based and fuzzy expert systems.

There are many more techniques to dealing with the cold-start problem by combining collaborative filtering with a content-based methods, including using simultaneous co-clustering [25], self-organizing maps, or Siamese neural networks [3]. There are also some attempts to combine RSs with various dimensionality reduction techniques [26]. Considering the discussed problem of missing or insufficient information, it seems interesting to refer to the dimensionality reduction methods based on the granularization of the attribute space [27], and particularly on resilient techniques [28], [29] - i.e., resistant to data deficiencies. The hybridization of soft computing techniques with collaborative and content-based methods is a wide-ranging field of research and an interesting area for the further development of recommendation systems [30], particularly interesting for context-aware RSs [31], [4].

Some approaches to dealing with cold-start refer to popularity measures, e.g., on the recent trend in users' preferences or always returning the most popular items [14], [10]. However, these may be very misleading and result in so-called popularity bias since users often differ in their preferences, which may also vary between types of products and their characteristics [18]. Hence, an additional effort to deal with biases in data is required [32]. Another interesting approach to dealing with insufficient or missing historical transactions avail additional

sources of information to enhance the data representation. In particular, in [21], the authors train RSs with the Linked Open Data model based on DBpedia to find enough information about new entities. When dealing with the cold-start problem, some researchers rely on directly inquiring the users about their preferences. Such information may be collected, e.g., via survey or by asking users to select the most relevant picture related to the desired item [33]. Combining community-based knowledge with association rule mining to alleviate the cold-start problem is also bringing very promising results [31]. Referring to association rule mining (cf. [34]) and frequent pattern mining (cf. [35]) techniques to address the cold-start problem is interesting also from the perspective of speeding up the recommender systems. For this reason, frequent patterns mining is particularly interesting in our research, and we review this field in detail in Section III-B.

III. PRELIMINARIES

In this section, we briefly summarize the academic knowledge of collaborative filtering and frequent pattern mining techniques. Then, we review some of the research literature related to addressing the cold-start problem.

A. Collaborative Filtering

The basic idea behind collaborative filtering (CF) is that the users who have similar preferences in the past tend to behave similarly in the future. Basically, CF-based methods rely only on users rating history to generate recommendations, meaning that the more ratings the users provide, the more accurate the recommendation become [4]. Usually, the historical ratings or preferences can be acquired explicitly or implicitly. So, the CF-based methods are often distinguished by whether they operate over explicit ratings, where the user explicitly rate particular items, or implicit ratings, where the ratings are inferred from observable user activity, such as products bought, songs heard, visited pages, or any other types of information access patterns [4]. In the literature, collaborative filtering methods can be classified into two main categories: (i) memory-based techniques, and (ii) model-based techniques.

The memory-based technique uses directly the rating history, which is stored in memory, to predict the rating of items that the user has not seen before. However, the memory-based techniques can be grouped into two different classes: (i) user-based collaborative filtering, and (ii) item-based collaborative filtering. The user-based collaborative filtering, also known as k-NN collaborative filtering, works by finding the other users (neighbors) whose historical rating behavior is similar to that of the target user and then using their top-rated products to predict what the target user will like [36]. To mathematically formulate the problem, let us assume there is a list of users $U = \{u_1, u_2, \dots, u_m\}$ and a list of items $I = \{i_1, i_2, \dots, i_n\}$. Then, the user item rating matrix consists of a set of ratings $v_{i,j}$ corresponding to the rating for user i on item j . If I_i is

the set of items on which user i has rated in the past, then we can define the average rating for user i as follows [36]:

$$\bar{v}_i = \frac{1}{|I_i|} \sum_{j \in I_i} v_{i,j} \quad (1)$$

In user-based collaborative filtering, we estimate the rating of item j that has not yet rated by the target user a as follows [36] [37]:

$$p_{a,j} = \bar{v}_a + \frac{\sum_{i=1}^k s(a,i)(v_{i,j} - \bar{v}_i)}{\sum_{i=1}^k |s(a,i)|} \quad (2)$$

where k is the number of most similar users (nearest neighbors) to a . The weights $s(a,i)$ can reflect the degree of similarity between each neighbor i and the target user a . On the other hand, item-based collaborative filtering is just an analogous procedure to the previous method. The similarity scores can also be used to generate predictions using a weighted average, similar to the procedure used in user-based collaborative filtering. Mathematically, we can predict the rating of item j that has not yet been rated by the target user a as follows [36] [37]:

$$p_{a,j} = \frac{\sum_{i=1}^k s(j,i)(v_{a,i})}{\sum_{i=1}^k |s(j,i)|} \quad (3)$$

where k is the number of most similar items (nearest neighbors) to j that the target user a has rated in the past. However, the most popular metrics used to calculate the similarity between different users, or items, are cosine similarity and Pearson correlation. Finally, the recommendations are generated by selecting the candidate items with the highest predictions.

On the other hand, the model-based technique works by learning a predictive model using the rating history. Basically, it is based on matrix factorization which uses the rating history to learn the latent preferences of users and items. Matrix factorization is an unsupervised learning method that is used for dimensionality reduction. One of the most popular techniques applied for dimensionality reduction is Singular Value Decomposition (SVD). Mathematically, let us assume M is the user item rating matrix. The SVD of M is the factorization of M into three constituent matrices such that [37]:

$$M = U\Sigma V^T \quad (4)$$

where U is an orthogonal matrix representing left singular vectors of M . V is an orthogonal matrix representing right singular vectors of M . Σ is a diagonal matrix whose values σ_i are the singular values of M [37].

B. Frequent Pattern Mining

The basic idea of frequent pattern mining, also known as association rule mining, is to search for all relationships between elements in a given massive dataset. It helps us to discover the associations among items using every distinct transaction in large databases. The key difference between association rules mining and collaborative filtering is that in

association rules mining we aim to find global or shared preferences across all users rather than finding an individual's preference like in collaborative filtering-based techniques [38] [39] [40].

At a basic level, association rule mining analyzes the dataset searching for frequent patterns (itemsets) using machine learning models. To define the previous problem mathematically, let $I = \{i_1, i_2, \dots, i_m\}$ be an itemset and let D be a set of transactions where each transaction T is a nonempty itemset such that $T \subseteq I$. An association rule is an implication of the form $A \Rightarrow B$, where $A \subset I$, $B \subset I$, $A \neq \emptyset$, $B \neq \emptyset$, $A \cap B = \emptyset$. In the rule $A \Rightarrow B$, A is called the antecedent and B is called the consequent. Various metrics are used to identify the most important itemset and calculate their strength, such as support, confidence, and lift. Support metric [40] is the measure that gives an idea of how frequent an itemset is in all transactions. In other words, the support metric represents the number of transactions that contain the itemset. The Equation 5 shows how we calculate the support for an association rule.

$$\text{support}(A \Rightarrow B) = P(A \cup B) \quad (5)$$

On the other hand, the confidence [40] indicates how often the rule is true. It defines the percentage of transactions containing the antecedent A that also contain the consequent B . It can be taken as the conditional probability as shown in Equation 6.

$$\text{confidence}(A \Rightarrow B) = P(B|A) = \frac{\text{support}(A \cup B)}{\text{support}(A)} \quad (6)$$

Finally, the lift is a correlation measure used to discover and exclude the weak rules that have high confidence. The Equation 7 shows that the lift measure is calculated by dividing the confidence by the unconditional probability of the consequent [40] [38].

$$\text{lift}(A \Rightarrow B) = \frac{P(A \cup B)}{P(A)P(B)} = \frac{\text{support}(A \cup B)}{\text{support}(A)\text{support}(B)} \quad (7)$$

If the lift value is equal to 1, then A and B are independent and there is no correlation between them. If the lift value is greater than 1, then A and B are positively correlated. If the lift value is less than 1, then A and B are negatively correlated.

Various algorithms exist for mining frequent itemsets, such as Apriori [39] [41], AprioriTID [39] [41], Apriori Hybrid [39] [41], and FP-growth (Frequent pattern) [39] [42]. In this paper, we employ FP-growth algorithm to generate frequent itemsets. What makes FP-growth better than other algorithms is the fact that FP-growth algorithm relies on FP-tree (frequent pattern tree) data structure to store all data concisely and compactly which greatly helps to avoid the candidate generation step. Moreover, once the FP-tree is constructed, we can directly use a recursive divide-and-conquer approach to efficiently mine the frequent itemsets without any need to scan the database over and over again like in other algorithms [42].

IV. FREQUENT PATTERN MINING FRAMEWORK FOR RECOMMENDER SYSTEMS (FPRS)

The main problem we address in this paper is to alleviate the impact of new users and new items cold-start in recommender systems based on collaborative filtering techniques. In theory, collaborative filtering methods can be grouped into two general categories (i) memory-based techniques and (ii) model-based techniques. In memory-based techniques, we calculate the similarities between users/items based on the rating history and then generate recommendations based on the most similar users/items. In model-based techniques, we rely, e.g., on matrix factorization methods to learn the latent factors of users and items and then decompose the user-item interaction (rating) matrix into the product of two lower dimensionality matrices. Collaborative filtering methods are strictly relying on user ratings or user interactions. For that reason, these methods suffer from the cold-start problem whenever a new user joins the system or when a new item is added. In practice, both situations often lead to the inability to provide accurate or meaningful recommendations.

To tackle the cold-start problem, we implement the Frequent Pattern mining framework for Recommender Systems (FPRS). This framework extends the popularity-based approach by employing frequent pattern mining techniques to learn the user preferences depending on users' and items' characteristics. Fig 1 shows the high-level design which is used to develop the FPRS framework. The process of generating the recommendations consists of four stages: (i) Data Input, (ii) Data Preparation, (iii) Frequent Pattern Mining, and (iv) Recommendation Generation. In the first stage, we enrich the user-item rating matrix by users' demographics and items' characteristics. The data preparation stage consists of three steps. In the first one, we store only the favorable reviews by filtering out every review/rating below a determined threshold. In the second step, we perform attributes analysis and check the validity of using them for generating the recommendation. In the last step, we split the dataset for each selected attribute. It is important to note that we follow multiple strategies to perform the attribute selection. More details about these strategies will be provided later in this section. Then, in the third stage, we generate frequent itemsets using FP-Growth algorithm. Finally, we produce the recommendations in the last stage for user cold-start and item cold-start. However, the FPRS framework consists of two main modules: (i) user cold-start module, and (ii) item cold-start module. Each of these modules has dedicated strategies that are used to select the features and produce the recommendations.

Strategy 1 User Cold-Start Module

- 1: Use entire training set (no records splitting)
 - 2: Generate frequent 1-itemsets {support > min_support}
 - 3: Recommend all these frequent 1-itemsets to any new user
-

a) user cold-start module: In this module, we focus on generating recommendations for new users who signed

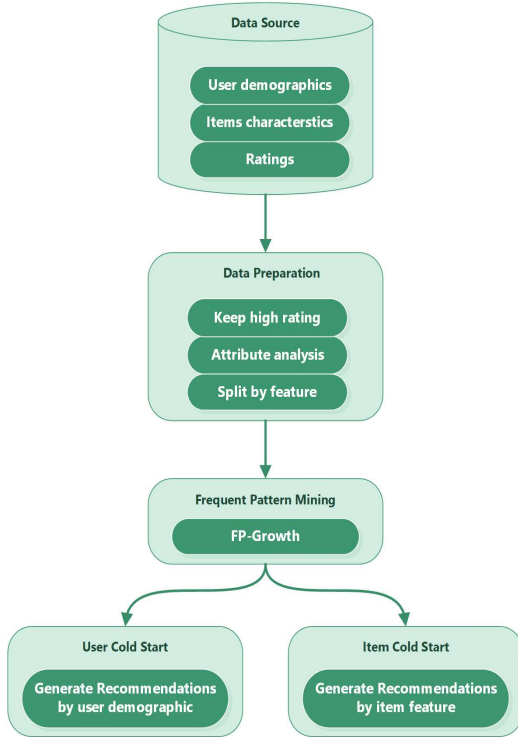


Fig. 1: Frequent Pattern Mining Framework For Recommender Systems

up recently to the system and most likely do not have, or have very few, ratings in the user-item rating matrix. We follow two strategies to generate such recommendations. These strategies differ in two main factors: (i) features selected to split the data, and (ii) the way how the frequent patterns are utilized to generate the recommendations. More details about the strategies followed in the user cold-start module are introduced in Strategy 1 and Strategy 2.

Strategy 2 User Cold-Start Module

- 1: Split the records based on users demographics (i.e. gender)
 - 2: Generate frequent 1-itemsets $\{\text{support} > \text{min_support}\}$
 - 3: Recommend to the new user all frequent 1-itemsets which are generated based on the demographics of new user
-

b) item cold-start module: In this module, we focus on generating recommendations for new items which are recently added to the system and most likely do not have, or have very few, ratings in the past. We follow multiple strategies to generate such recommendations. More details about the strategies followed in the item cold-start module are provided in Strategy 3, Strategy 4, and Strategy 5.

Finally, it is worth mentioning that the threshold values used in the above strategies are selected carefully by objectively searching for a good set of values that achieves the best performance on a given dataset. More details on how we choose these values are provided in Section V.

Strategy 3 Item Cold-Start Module

- 1: Split the records based on items characteristics (i.e. genre)
 - 2: Generate frequent itemsets $\{\text{support} > \text{min_support}\}$
 - 3: Set the participation percentage threshold
 - 4: **for each** value in genre **do**
 - 5: Find all users who involved in creating larger than the participation threshold of frequent itemsets
 - 6: **end for**
 - 7: Recommend the new item based on its genre to all users found in previous step
-

Strategy 4 Item Cold-Start Module

- 1: Split the records based on users demographics and items characteristics (i.e. gender and genre)
 - 2: Generate frequent itemsets $\{\text{support} > \text{min_support}\}$
 - 3: Set the participation percentage threshold
 - 4: **for each** value in genre **do**
 - 5: Find the dominant gender by counting how many frequent itemsets are generated by male and female
 - 6: **end for**
 - 7: Recommend the new item based on its genre to all users belong to dominant gender who involved in creating larger than the participation threshold of frequent itemsets
-

Strategy 5 Item Cold-Start Module

- 1: Split the records based on users demographics and items characteristics (i.e. gender and genre)
 - 2: Generate frequent 1-itemsets $\{\text{support} > \text{min_support}\}$
 - 3: Set the participation percentage threshold
 - 4: Assign frequent 1-itemsets created by male to one cluster, and frequent 1-itemsets created by female to another cluster
 - 5: Find the center of each cluster
 - 6: Calculate the distance between the new item and the center of each cluster
 - 7: Recommend the new item to all users who involved in creating larger than the participation threshold of frequent itemsets in the closest cluster
-

V. EVALUATION METHODOLOGY

In this section, we conduct comprehensive experiments to evaluate the performance of the FPRS recommender system.

A. Dataset and Evaluation Measures

In our experiments, we used two datasets (MovieLens 100K and MovieLens 1M)¹ which were collected by the GroupLens research project at the University of Minnesota. MovieLens 100K contains 100,000 ratings given by 943 users on 1682 movies on a scale from 1 to 5. While MovieLens 1M contains 1,000,000 ratings of approximately 3,900 movies made by 6,040 users on a scale from 1 to 5. In both datasets, we

¹<https://grouplens.org/datasets/movielens/>

combine three files (users.data, items.data, ratings.data) in order to join users' demographics, items' characteristics, and ratings in one dataset. The final/joined dataset contains userId, itemId, rating, gender, age, occupation, and genres attributes (cf. Table I). Moreover, we performed further analysis of the features we used in our experiments (gender, genre) to understand the interrelation between these features and obtained results. Figures 2 show the most popular movie genres among males and females for both datasets (MovieLens 100K and MovieLens 1M).

TABLE I: Selected data characteristics.

Attribute Name	Data Type	Value Range (MovieLens 100K)	Value Range (MovieLens 1M)
gender	Character	M-F	M-F
age	Number	Under 18-73	Under 18-56
occupation	Text	21 occupation	21 occupation
genres	Text	19 genres	19 genres

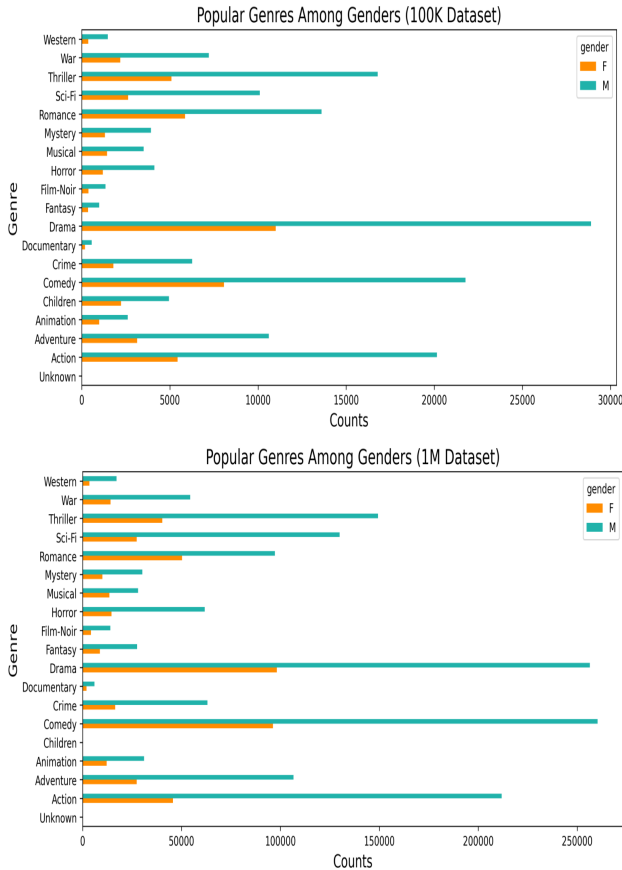


Fig. 2: Histogram of the variables (rating, genre and gender) in the MovieLens 100K and 1M datasets.

After clearing the data from invalid records, we split it into training and testing datasets according to FPRS module as follows:

- For the user cold-start module testing, the 20 users with the highest number of ratings on the 50 most popular movies were selected. The ratings given by all those 20 users (9052 records in MovieLens 100K and 25533 records in MovieLens 1M) are considered a testing set, keeping the rest of the records in the training set. This way, all the record related to the selected users were removed from the test set, which simulates the cold-start problem associated with new users.
- To properly evaluate the item cold-start module, the testing data is chosen similarly. We firstly find the 50 most active users. Then, we select the 20 most rated movies by those 50 users. The ratings of all those 20 movies by all the users in our data (7320 records in MovieLens 100K and 41105 records in MovieLens 1M) are considered as a testing set, keeping the rest of the records in the training set. Note that all the ratings for the selected movies are removed from the training data set, which corresponds to the item cold-start.

In our study, we consider a binary decision task whether a given item (i.e., movie) is appropriate for the user. To correctly model this situation for the MovieLens data, we assume that films rated by users 4 or 5 are preferred by them (belong to the positive class). In contrast, those ranked lower are poorly matched to the users. Therefore, the FPRS recommender system feedback for each new user or item is binary information: recommend or not recommend. Following that, in order to assess the quality of the prediction, the F1 measure is used [43].

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \quad (8)$$

where precision quantifies the number of correct positive recommendations made (see Equation 9). While recall quantifies the number of correct positive recommendations made out of all positive predictions that could have been made (see Equation 10).

$$\text{Precision} = \frac{TP}{TP + FP} \quad (9)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (10)$$

Moreover, we use the accuracy metric to measure all the correctly identified cases. This measure is mostly used when all the classes are equally important.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \quad (11)$$

B. Baseline Recommender System

To showcase the strengths of frequent pattern mining in RS, we build a baseline model in a similar way to the strategies described previously, but the FP-growth algorithm was omitted. In order to evaluate both modules of FPRS, two versions of baseline RS are developed as follows:

- New user baseline model: for each movie in the training set, the most common gender and age group of the user was found. When a new user comes, they get recommended all movies that were assigned to their age group and gender.
- New item baseline model: it works similarly. We find the most popular (watched) genre for each user. Then each new movie is recommended to all users whose favorite genre was the same as the new movie's.

C. Performance Comparison and Analysis

In order to provide a fair comparison, we use precision, recall, F1, and accuracy measures to compare the performance of FPRS against the baseline RS. After splitting the dataset into the training and testing sets and training both baseline and FPRS recommendation systems, we run two experiments to evaluate the user cold-start module and item cold-start module.

a) *User cold-start module*: In the performed experiment, we evaluated the user cold-start module in FPRS. We calculated precision, recall, F1, and accuracy measures for the results generated by the baseline RS and FPRS following the two strategies, which we have described in Section IV. The comparison results of this evaluation method are shown in Tables II and III. The results show that all developed strategies outperformed the baseline RS for precision, F1, and accuracy on both data-sets. However, the baseline solution reported higher recall, which is quite natural since the developed methods are more selective, providing more apt results with a tradeoff that some potentially relevant movies may be omitted. For the user-cold start problem, both strategies were evaluated at $min_support$ value of 0.08 and performed similarly. The second one was just slightly better.

TABLE II: Evaluation for user cold-start (MovieLens 100k).

Strategy	Precision	Recall	F1-score	Accuracy
Baseline	0.24	0.72	0.33	0.44
Strategy 1	0.58	0.56	0.57	0.80
Strategy 2	0.59	0.58	0.58	0.80

TABLE III: Evaluation for user cold-start (MovieLens 1M).

Strategy	Precision	Recall	F1-score	Accuracy
Baseline	0.33	0.82	0.44	0.47
Strategy 1	0.63	0.63	0.62	0.77
Strategy 2	0.64	0.64	0.63	0.78

b) *Item cold-start module*: In the second experiment, we evaluated the item cold-start module in FPRS. We calculated precision, recall, F1, and accuracy measures for the results generated by the baseline RS and FPRS following all the strategies described in Section IV. The comparative summary of this evaluation is shown in Tables II and III. The results show that the performance of FPRS, using all strategies, is superior to the baseline solution. However, the results differ

slightly between datasets. For MovieLens 100K, all strategies reported similar recall. Regarding precision and F1 measures, the most successful in dealing with new items in this data appeared to be strategy no. 4, which is based on both items' and users' characteristics. However, for the applications that do require high accuracy, it would be better to apply strategy no. 5, which was also superior in terms of recall, F1, and accuracy on the second data-set (MovieLens 1M). All strategies were evaluated at the participation threshold value of 30% and $min_support$ value of 0.2.

TABLE IV: Evaluation for item cold-start (MovieLens 100k).

Strategy	Precision	Recall	F1-score	Accuracy
Baseline	0.38	0.3	0.32	0.59
Strategy 3	0.69	0.64	0.66	0.75
Strategy 4	0.79	0.62	0.69	0.84
Strategy 5	0.67	0.63	0.63	0.86

TABLE V: Evaluation for item cold-start (MovieLens 1M).

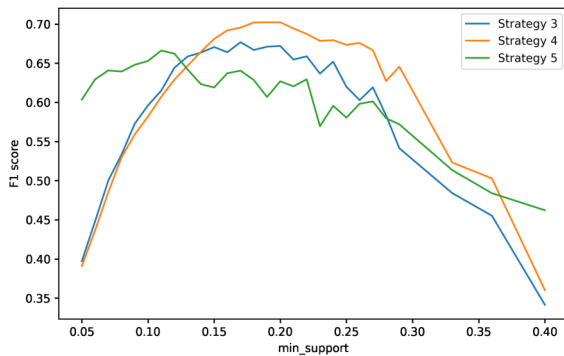
Strategy	Precision	Recall	F1-score	Accuracy
Baseline	0.49	0.43	0.43	0.64
Strategy 3	0.65	0.71	0.64	0.76
Strategy 4	0.64	0.73	0.66	0.83
Strategy 5	0.6	0.79	0.66	0.86

D. Thresholds Sensitivity Analysis

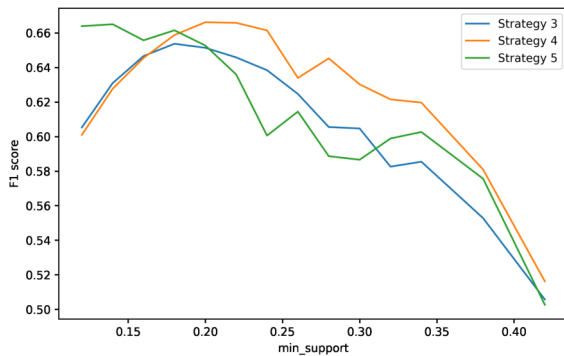
In FPRS model, we use some threshold values, such as $min_support$ and participation percentage, in order to extract frequent itemsets and produce relevant recommendations for new users and new items. In this section, we conduct some experiments to show how changing those values may impact performance of FPRS. Moreover, the output of this experiment helps to find the optimal values of these thresholds, and hence to conduct fair and reliable experiments.

In the first experiment, we aim to find the optimal value of min_sup threshold by evaluating FPRS (item cold-start module) using different min_sup values. Fig 3a shows how the F1-score of FPRS is impacted by applying different values for MovieLena 100K data. Observably, the best min_sup values for all strategies used in FPRS (item cold-start module) are between 0.1 and 0.2. The similar observations, regarding MovieLens 1M data, we may find in Figure 3b.

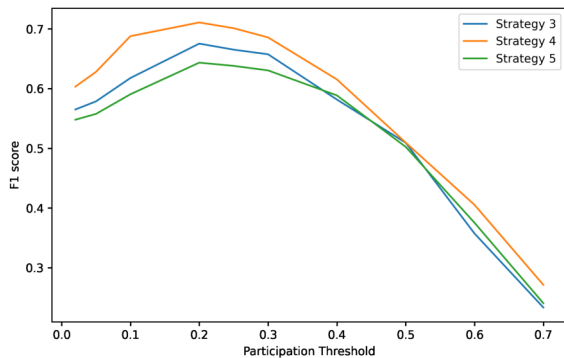
In the second experiment, we search for the optimal value of the participation threshold in FPRS (item cold-start module). Figures 3c and 3d show how F1-score of FPRS is impacted by applying different values for both investigated datasets. Observably, the best participation threshold values for all strategies used in FPRS (item cold-start module) are between 15% and 30%. Finally, it is worth noting that when we run this experiment, we use the optimal value of min_sup we found in the previous experiment.



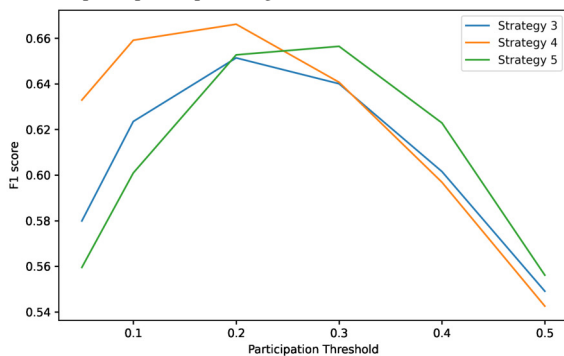
(a) min_supp threshold (MovieLens 100k)



(b) min_supp threshold (MovieLens 1M)



(c) participation percentage threshold (MovieLens 100k)



(d) participation percentage threshold (MovieLens 1M)

Fig. 3: Sensitivity analysis for min_supp and participation percentage thresholds for MovieLens 100k and 1M.

VI. CONCLUSIONS AND FUTURE WORKS

This article introduces FPRS, a novel recommender system, which methodically utilizes the ratings to discover frequent itemsets associated with selected users/items features and then incorporates these frequent itemsets in generating recommendations for new users and items. Our study evaluates multiple strategies for creating frequent itemsets to produce meaningful and relevant recommendations.

To evaluate FPRS, we conducted comprehensive experiments on MovieLens 100K and 1M datasets using the FP-growth algorithm to generate the frequent itemsets. The experimental results show that FPRS has outperformed the baseline recommender system in terms of the precision, recall, F1, and accuracy measures.

In the future work, we plan to incorporate additional contextual information and evaluate more advanced algorithms, such as AprioriTID and Apriori Hybrid, in the process of producing frequent patterns. Another important aspect to consider is to evaluate our method against more state-of-the-art recommender systems on various datasets. Furthermore, we plan to consider changes in users' behavior and preferences by periodically updating frequent itemsets based on recent changes in rating history. It would also be of value to extend the users' and items' data representation by applying a more advanced feature extraction to model the similarities among them more effectively [44], [45], [46].

ACKNOWLEDGMENT

Research co-funded by Polish National Science Centre (NCN) grant no. 2018/31/N/ST6/00610.

REFERENCES

- [1] Z. Batmaz, A. Yürekli, A. Bilge, and C. Kaleli, "A review on deep learning for recommender systems: challenges and remedies," *Artif. Intell. Rev.*, vol. 52, no. 1, pp. 1–37, 2019. doi: 10.1007/s10462-018-9654-y
- [2] M. Kawai, H. Sato, and T. Shiohama, "Topic model-based recommender systems and their applications to cold-start problems," *Expert Systems with Applications*, vol. 202, p. 117129, 2022. doi: 10.1016/j.eswa.2022.117129
- [3] M. Pulis and J. Bajada, *Siamese Neural Networks for Content-Based Cold-Start Music Recommendation*. New York, NY, USA: Association for Computing Machinery, 2021, p. 719–723. ISBN 9781450384582
- [4] E. Kannout, "Context clustering-based recommender systems," in *2020 15th Conference on Computer Science and Information Systems (FedCSIS)*, 2020. doi: 10.15439/2020F54 pp. 85–91.
- [5] M. Pondel and J. Korczak, "Collective clustering of marketing data - recommendation system upsally," in *Proceedings of the 2018 Federated Conference on Computer Science and Information Systems, FedCSIS 2018, Poznań, Poland, September 9-12, 2018*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 15, 2018. doi: 10.15439/2018F217 pp. 801–810.
- [6] B. Yi, X. Shen, H. Liu, Z. Zhang, W. Zhang, S. Liu, and N. Xiong, "Deep Matrix Factorization With Implicit Feedback Embedding for Recommendation System," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 8, pp. 4591–4601, 2019. doi: 10.1109/TII.2019.2893714
- [7] M. Jalili, S. Ahmadian, M. Izadi, P. Moradi, and M. Salehi, "Evaluating Collaborative Filtering Recommender Algorithms: A Survey," *IEEE Access*, vol. 6, pp. 74 003–74 024, 2018. doi: 10.1109/ACCESS.2018.2883742

- [8] R. Chen, Q. Hua, Y.-S. Chang, B. Wang, L. Zhang, and X. Kong, "A Survey of Collaborative Filtering-Based Recommender Systems: From Traditional Methods to Hybrid Methods Based on Social Networks," *IEEE Access*, vol. 6, pp. 64 301–64 320, 2018. doi: 10.1109/ACCESS.2018.2877208
- [9] B. Walek and V. Fojtik, "A hybrid recommender system for recommending relevant movies using an expert system," *Expert Systems with Applications*, vol. 158, p. 113452, 2020. doi: 10.1016/j.eswa.2020.113452
- [10] Y. Pérez-Almaguer, R. Yera, A. A. Alzahrani, and L. Martínez, "Content-based group recommender systems: A general taxonomy and further improvements," *Expert Systems with Applications*, vol. 184, p. 115444, 2021. doi: 10.1016/j.eswa.2021.115444
- [11] Y. Afoudi, M. Lazaar, and M. Al Achhab, "Hybrid recommendation system combined content-based filtering and collaborative prediction using artificial neural network," *Simulation Modelling Practice and Theory*, vol. 113, p. 102375, 2021. doi: 10.1016/j.simpat.2021.102375
- [12] R. Pasricha and J. McAuley, "Translation-Based Factorization Machines for Sequential Recommendation," in *Proceedings of the 12th ACM Conference on Recommender Systems*, ser. RecSys'18. New York, NY, USA: Association for Computing Machinery, 2018. doi: 10.1145/3240323.3240356. ISBN 9781450359016 p. 63–71.
- [13] H. Wu, Z. Zhang, K. Yue, B. Zhang, J. He, and L. Sun, "Dual-regularized matrix factorization with deep neural networks for recommender systems," *Knowledge-Based Systems*, vol. 145, pp. 46–58, 2018. doi: 10.1016/j.knsys.2018.01.003
- [14] F. Liu, R. Tang, X. Li, W. Zhang, Y. Ye, H. Chen, H. Guo, and Y. Zhang, "Deep Reinforcement Learning based Recommendation with Explicit User-Item Interactions Modeling," 2018.
- [15] M. K. Najafabadi, A. H. Mohamed, and M. N. Mahrin, "A survey on data mining techniques in recommender systems," *Soft Comput.*, vol. 23, no. 2, pp. 627–654, 2019. doi: 10.1007/s00500-017-2918-7
- [16] M. Grzegorowski, A. Janusz, S. Lazewski, M. Swiechowski, and M. Jankowska, "Prescriptive analytics for optimization of FMCG delivery plans," in *Information Processing and Management of Uncertainty in Knowledge-Based Systems - 19th International Conference, IPMU 2022, Milan, Italy, July 11-15, 2022, Proceedings, Part II*, ser. Communications in Computer and Information Science, D. Ciucci, I. Couso, J. Medina, D. Ślęzak, D. Petturiti, B. Bouchon-Meunier, and R. R. Yager, Eds., vol. 1602. Springer, 2022. doi: 10.1007/978-3-031-08974-9_4 pp. 44–53.
- [17] A. Janusz, M. Grzegorowski, M. Michalak, E. Wróbel, M. Sikora, and D. Ślęzak, "Predicting Seismic Events in Coal Mines Based on Underground Sensor Measurements," *Engineering Applications of Artificial Intelligence*, vol. 64, pp. 83–94, 2017.
- [18] Z. Zhu, J. Kim, T. Nguyen, A. Fenton, and J. Caverlee, *Fairness among New Items in Cold Start Recommender Systems*. New York, NY, USA: Association for Computing Machinery, 2021, p. 767–776. ISBN 9781450380379
- [19] Z. Cui, X. Xu, F. XUE, X. Cai, Y. Cao, W. Zhang, and J. Chen, "Personalized Recommendation System Based on Collaborative Filtering for IoT Scenarios," *IEEE Transactions on Services Computing*, vol. 13, no. 4, pp. 685–695, 2020. doi: 10.1109/TSC.2020.2964552
- [20] A. Karpus, I. Vagliano, K. Goczyla, and M. Morisio, "An ontology-based contextual pre-filtering technique for recommender systems," in *Proceedings of the 2016 Federated Conference on Computer Science and Information Systems, FedCSIS 2016, Gdańsk, Poland, September 11-14, 2016*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 8. IEEE, 2016. doi: 10.15439/2016F428 pp. 411–420.
- [21] S. Natarajan, S. Vairavasundaram, S. Natarajan, and A. H. Gandomi, "Resolving data sparsity and cold start problem in collaborative filtering recommender system using linked open data," *Expert Systems with Applications*, vol. 149, p. 113248, 2020. doi: 10.1016/j.eswa.2020.113248
- [22] A. Karpus, I. Vagliano, and K. Goczyla, "Serendipitous recommendations through ontology-based contextual pre-filtering," in *Beyond Databases, Architectures and Structures. Towards Efficient Solutions for Data Analysis and Knowledge Representation - 13th International Conference, BDAS 2017, Ustroń, Poland, May 30 - June 2, 2017, Proceedings*, ser. Communications in Computer and Information Science, S. Kozielski, D. Mrozek, P. Kasproski, B. Malysiak-Mrozek, and D. Kostrzewa, Eds., vol. 716, 2017. doi: 10.1007/978-3-319-58274-0_21 pp. 246–259.
- [23] A. Pawlicka, M. Pawlicki, R. Kozik, and R. S. Choraś, "A systematic review of recommender systems and their applications in cybersecurity," *Sensors*, vol. 21, no. 15, 2021. doi: 10.3390/s21155248. [Online]. Available: <https://www.mdpi.com/1424-8220/21/15/5248>
- [24] K. Ghazinour, S. Matwin, and M. Sokolova, "Monitoring and recommending privacy settings in social networks," in *Joint 2013 EDBT/ICDT Conferences, EDBT/ICDT '13, Genoa, Italy, March 22, 2013, Workshop Proceedings*, G. Guerrini, Ed. ACM, 2013. doi: 10.1145/2457317.2457344 pp. 164–168.
- [25] A. L. Vazine Pereira and E. R. Hruschka, "Simultaneous co-clustering and learning to address the cold start problem in recommender systems," *Knowledge-Based Systems*, vol. 82, pp. 11–19, 2015. doi: 10.1016/j.knsys.2015.02.016
- [26] M. Nilashi, O. Ibrahim, and K. Bagherifard, "A recommender system based on collaborative filtering using ontology and dimensionality reduction techniques," *Expert Systems with Applications*, vol. 92, pp. 507–520, 2018. doi: 10.1016/j.eswa.2017.09.058
- [27] M. Grzegorowski, A. Janusz, D. Ślęzak, and M. S. Szczuka, "On the role of feature space granulation in feature selection processes," in *2017 IEEE International Conference on Big Data (IEEE BigData 2017), Boston, MA, USA, December 11-14, 2017*, J. Nie, Z. Obradovic, T. Suzumura, R. Ghosh, R. Nambiar, C. Wang, H. Zang, R. Baeza-Yates, X. Hu, J. Kepner, A. Cuzzocrea, J. Tang, and M. Toyoda, Eds. IEEE Computer Society, 2017. doi: 10.1109/BigData.2017.8258124 pp. 1806–1815.
- [28] M. Grzegorowski and D. Ślęzak, "On resilient feature selection: Computational foundations of r-C-reducts," *Inf. Sci.*, vol. 499, pp. 25–44, 2019. doi: 10.1016/j.ins.2019.05.041
- [29] M. Grzegorowski, "Governance of the Redundancy in the Feature Selection Based on Rough Sets' Reducts," in *Rough Sets - International Joint Conference, IJCRS 2016, Santiago de Chile, Chile, October 7-11, 2016, Proceedings*, ser. Lecture Notes in Computer Science, V. Flores, F. A. C. Gomide, A. Janusz, C. Meneses, D. Miao, G. Peters, D. Ślęzak, G. Wang, R. Weber, and Y. Yao, Eds., vol. 9920, 2016. doi: 10.1007/978-3-319-47160-0_50 pp. 548–557. [Online]. Available: https://doi.org/10.1007/978-3-319-47160-0_50
- [30] M. asid and R. Ali, *Use of Soft Computing Techniques for Recommender Systems: An Overview*. Singapore: Springer Singapore, 2017, pp. 61–80. ISBN 978-981-10-7098-3
- [31] I. Viktoratos, A. Tsadiras, and N. Bassiliades, "Combining community-based knowledge with association rule mining to alleviate the cold start problem in context-aware recommender systems," *Expert Systems with Applications*, vol. 101, pp. 78–90, 2018. doi: doi.org/10.1016/j.eswa.2018.01.044
- [32] H. Steck, "Collaborative filtering via high-dimensional regression," *CoRR*, vol. abs/1904.13033, 2019. doi: 10.48550/ARXIV.1904.13033
- [33] H. Kwasnicka and T. Ovedenski, "Pix2trips - a system supporting small groups of urban tourists," in *Proceedings of the 16th Conference on Computer Science and Intelligence Systems, Online, September 2-5, 2021*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, M. Paprzycki, and D. Ślęzak, Eds., vol. 25, 2021. doi: 10.15439/2021F130 pp. 141–145.
- [34] H. Sobhanan and A. K. Mariappan, "Addressing cold start problem in recommender systems using association rules and clustering technique," in *2013 International Conference on Computer Communication and Informatics*, 2013. doi: 10.1109/ICCCI.2013.6466121 pp. 1–5.
- [35] W. Feng, Q. Zhu, J. Zhuang, and S. Yu, "An expert recommendation algorithm based on pearson correlation coefficient and fp-growth," *Clust. Comput.*, vol. 22, no. Supplement, pp. 7401–7412, 2019. doi: 10.1007/s10586-017-1576-y
- [36] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," in *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, ser. UAI'98. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1998. ISBN 155860555X p. 43–52.
- [37] M. D. Ekstrand, J. T. Riedl, and J. A. Konstan, "Collaborative filtering recommender systems," *Found. Trends Hum.-Comput. Interact.*, vol. 4, no. 2, p. 81–173, feb 2011. doi: 10.1561/1100000009. [Online]. Available: <https://doi.org/10.1561/1100000009>
- [38] J. Han, M. Kamber, and J. Pei, "6 - mining frequent patterns, associations, and correlations: Basic concepts and methods," in *Data Mining (Third Edition)*, third edition ed., ser. The Morgan Kaufmann Series in Data Management Systems, J. Han, M. Kamber, and J. Pei, Eds. Boston: Morgan Kaufmann, 2012, pp. 243–278. ISBN 978-0-12-381479-1. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/B978012381479100006X>

- [39] J. Hipp, U. Güntzer, and G. Nakhaeizadeh, "Algorithms for association rule mining — a general survey and comparison," *SIGKDD Explor. Newsl.*, vol. 2, no. 1, p. 58–64, jun 2000. doi: 10.1145/360402.360421. [Online]. Available: <https://doi.org/10.1145/360402.360421>
- [40] U. Fayyad, *Knowledge Discovery in Databases: An Overview*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 28–47. ISBN 978-3-662-04599-2. [Online]. Available: https://doi.org/10.1007/978-3-662-04599-2_2
- [41] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules in large databases," in *Proceedings of the 20th International Conference on Very Large Data Bases*, ser. VLDB '94. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1994. ISBN 1558601538 p. 487–499.
- [42] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," *SIGMOD Rec.*, vol. 29, no. 2, p. 1–12, may 2000. doi: 10.1145/335191.335372. [Online]. Available: <https://doi.org/10.1145/335191.335372>
- [43] T. Silveira, M. Zhang, X. Lin, Y. Liu, and S. Ma, "How good your recommender system is? A survey on evaluations in recommendation," *International Journal of Machine Learning and Cybernetics*, vol. 10, pp. 813–831, 2019. doi: 10.1007/s13042-017-0762-9
- [44] M. Grzegorowski, "Massively Parallel Feature Extraction Framework Application in Predicting Dangerous Seismic Events," in *Proceedings of the 2016 Federated Conference on Computer Science and Information Systems, FedCSIS 2016, Gdańsk, Poland, September 11-14, 2016*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 8. IEEE, 2016. doi: 10.15439/2016F90 pp. 225–229. [Online]. Available: <https://doi.org/10.15439/2016F90>
- [45] E. Zdravevski, P. Lameski, R. Mingov, A. Kulakov, and D. Gjorgievikj, "Robust histogram-based feature engineering of time series data," in *2015 Federated Conference on Computer Science and Information Systems, FedCSIS 2015, Łódź, Poland, September 13-16, 2015*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 5. IEEE, 2015. doi: 10.15439/2015F420 pp. 381–388. [Online]. Available: <https://doi.org/10.15439/2015F420>
- [46] M. Grzegorowski and S. Stawicki, "Window-based feature extraction framework for multi-sensor data: A posture recognition case study," in *2015 Federated Conference on Computer Science and Information Systems, FedCSIS 2015, Łódź, Poland, September 13-16, 2015*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 5. IEEE, 2015. doi: 10.15439/2015F425 pp. 397–405.

Learning edge importance in bipartite graph-based recommendations

Robert Kwiecieński

Faculty of Mathematics
and Computer Science

Adam Mickiewicz University
Uniwersytetu Poznańskiego 4
61-614 Poznań, Poland and

OLX Group

ul. Królowej Jadwigi 43,

61-872 Poznań, Poland

Email: r.kwiecinskipl@gmail.com

Tomasz Górecki

Faculty of Mathematics
and Computer Science

Adam Mickiewicz University
Uniwersytetu Poznańskiego 4
61-614 Poznań, Poland

Email: tomasz.gorecki@amu.edu.pl

Agata Filipowska

Poznań University of Economics and Business

Al. Niepodległości 10,
61-875 Poznań, Poland

and

OLX Group

ul. Królowej Jadwigi 43,

61-872 Poznań, Poland

Email: agata.filipowska@ue.poznan.pl

Abstract—In this work, we propose the P3 Learning to Rank (P3LTR) model, a generalization of the RP3Beta graph-based recommendation method. In our approach, we learn the importance of user-item relations based on features that are usually available in online recommendations (such as types of user-item past interactions and timestamps). We keep the simplicity and explainability of RP3Beta predictions. We report the improvements of P3LTR over RP3Beta on the OLX Jobs Interactions dataset, which we published.

I. INTRODUCTION

GRAPH-BASED RP3Beta model [1] is a very strong baseline on multiple recommender systems datasets [2], [3], [4]. This relatively simple model outperformed other approaches on our published OLX Jobs Interactions dataset and is currently a state-of-the-art collaborative filtering recommender system at OLX. In this work, we propose P3LTR (P3 Learning to Rank) model which generalizes the RP3Beta model.

In RP3Beta each user and item is represented as a node of the user-item bipartite graph. The recommendations are generated based on the paths of length 3 starting from a given user. The scores of these paths are calculated based on the scores assigned to the edges of the graph. Scores of the edges are directly calculated based on the degrees of the connected nodes. Hence, there is no learning process in this approach.

In P3LTR, to better leverage the importance of the user-item relation, we learn the score of a given edge based on the features of this edge. As features, we not only use node degrees but also utilize the sequence of interactions between two given nodes. It enables us to incorporate the information that the user visited the item several times and that the user not only clicked but applied for a given job, or how recent the click was.

In this work, we propose a training procedure and a loss function for the P3LTR model. We tune, train, and evaluate RP3Beta and P3LTR models on the OLX Jobs Interactions dataset.

The paper consists of 6 sections. The second section presents a literature review and formulates a research gap addressed by this work. Section III proposes the P3LTR model and describes its advantages and relation to the RP3Beta model. Section IV describes the considered dataset and hyperparameter tuning procedure. The results of our model are discussed in Section V. Section VI presents the conclusions and future perspectives.

II. RELATED WORKS

A. Recommender systems

Most digital platforms provide more choices than the user can explore in a reasonable time. Even a perfect search engine can not resolve this problem, because it requires users to know what they are looking for and to spend time providing this information. For this reason, powerful recommendation systems are developed by multiple companies, such as Netflix [5] or Amazon [6].

We usually distinguish two categories of recommendation methods: *content-based* and *collaborative filtering*. In *content-based* models [7], [8] we utilize user and item features to provide recommendations. The history of interactions between users and items is considered from the perspective of a single user. In contrast, *collaborative filtering* techniques [9], [10] do not consider additional information about users or items but utilize the rating history of all users at the same time to provide recommendations. During the last few decades several collaborative filtering recommendation techniques have been proposed: neighborhood-based (e.g., [11], [12]), matrix factorization-based (e.g., [13], [14]), graph-based (e.g., [1], [15] or Word2Vec-based (e.g., [16], [17])).

Another category of recommendation systems, *context-aware recommendation systems* (CARS) [18], [19], utilize contextual information of user-item interactions such as time or location. We can distinguish an important subcategory of these methods, *sequence-aware recommendation systems* [20], which utilizes sequentially-ordered user-item interaction logs.

In this work, we extend a graph-based collaborative-filtering approach that does not utilize additional contextual information. Our approach is a sequence-aware recommendation method that utilizes timestamps and types of interactions.

B. Graph-based recommendations

Many graph-based recommender systems are focused on producing the item and/or user embeddings. Some of them utilize the graph structure to produce random walks which are used as an input for the model which produces embeddings. For instance, Node2vec [21], or DeepWalk [22] utilize a SkipGram model [23]. In recent years, several collaborative filtering methods based on graph convolutional neural networks have been proposed [15], [24], [25], [26].

Another type of graph-based recommendation systems directly utilizes the graph structure to calculate the scores of items, usually by utilizing a user-item bipartite graph. Cooper *et al.* [27] proposed simple and efficient P3 and P3alpha methods which outperformed more complex and computationally demanding techniques [28], [29], [30]. Paudel *et al.* [1] extended this work by proposing the RP3Beta model, an extension of P3alpha which recommends popular items less often. These methods were recently used as a benchmark by Dacrema *et al.* [2], [3] and Anelli *et al.* [4] who compared them with several state-of-the-art neural recommendation methods. P3Alpha and RP3Beta demonstrated a very good performance against other baselines and neural models. The RP3Beta model provided the most accurate recommendations on some of the considered datasets (i.e., Pinterest [31], CiteULike-a [32] and MovieLens1M [33]; on MovieLens1M authors used their own random splits). In our previous work, as yet unpublished, we showed that RP3Beta outperforms other methods on the OLX Jobs Interactions dataset. It is currently the state-of-the-art collaborative-filtering recommendations technique deployed at OLX Jobs.

C. Research gap

The RP3Beta model does not utilize any information about user-item relations. Additionally, there is no learning process that could optimize model parameters. Hence it is not possible to learn the importance of edges in the user-item bipartite graph. In this work, we fill this gap by proposing a machine learning model which generalizes RP3Beta.

III. PROPOSED METHOD

A. Model

Let \mathcal{U} be a set of users and \mathcal{I} a set of items. For each user $u \in \mathcal{U}$ and item $i \in \mathcal{I}$ let r_{ui} be the score assigned by the model to the pair (u, i) . We denote the matrix of all user-item scores by \mathbf{R} .

We represent users and items as the nodes of the bipartite graph, where edges represent interactions between users and items. Let $\mathcal{N}(x)$ represent the set of nodes connected with the node x .

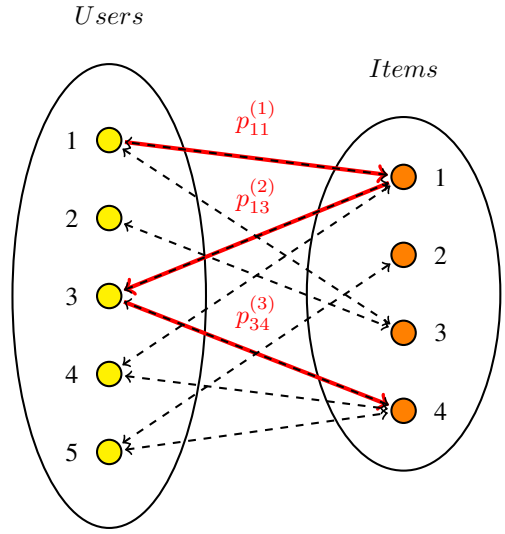


Fig. 1. Path of length 3 with edge scores. The path is highlighted by bold red line. Dashed lines represent the interactions between users and items.

For a given user $u \in \mathcal{U}$ our model recommends the items with the highest score r_{ui} , excluding the items which the user interacted with.

The score r_{ui} is calculated as the sum of the scores assigned to the paths of length 3 connecting u and i , i.e.:

$$r_{ui} = \sum_{i' \in \mathcal{N}(u)} \sum_{u' \in \mathcal{N}(i')} p(u, i', u', i),$$

where $p(u, i', u', i)$ is the score assigned to the given path. Following the idea used in the RP3Beta model, we factorize this score as:

$$p(u, i', u', i) = p_{ui'}^{(1)} p_{i'u'}^{(2)} p_{u'i}^{(3)},$$

where $p_{xy}^{(k)}$ is the score assigned to the edge connecting nodes x and y in the k -th layer, $k = 1, 2, 3$. The edge scores of a given path are illustrated in Fig. 1. With this assumption, the calculation of the scores is simplified in the following way:

$$\mathbf{R} = \mathbf{P}^{(1)} \mathbf{P}^{(2)} \mathbf{P}^{(3)},$$

where $\mathbf{P}^{(k)} = (p_{xy}^{(k)})$, $\mathbf{P}^{(1)}, \mathbf{P}^{(3)}$ are $|\mathcal{U}| \times |\mathcal{I}|$ matrices and $\mathbf{P}^{(2)}$ is $|\mathcal{I}| \times |\mathcal{U}|$ matrix.

In this work we propose to calculate the edge scores as the function of node features f_n and edge features f_e , i.e.:

$$p_{xy}^{(k)} = \phi^{(k)}(f_x^n, f_y^n, f_{xy}^e),$$

where f_x^n is feature vector of node x , f_{xy}^e is a feature vector of the edge connecting nodes x and y and $\phi^{(k)}$ can be any real-valued function (e.g., neural network). We will call the functions $\phi^{(k)}$ **feature encoders** and propose them below.

Assume that for each (user, item) pair we know the type of interactions between them with corresponding timestamps. Let $\mathcal{E} = \{e_1, e_2, \dots, e_{|\mathcal{E}|}\}$ be a set of all possible types of interactions (e.g., click, reply, purchase).

Then we define the following features:

- $\text{deg}(x)$ – degree of the node x (number of distinct users/items which interacted with x),
- $\text{rec}(x, y)$ – number of days which passed between the most recent user (x or y) interaction with the item (y or x) and the most recent interaction of this user with any item,
- $\text{ev}(e_i, x, y)$ – number of interactions of type e_i between x and y ,
- $\text{ev}(x, y)$ – number of interactions between x and y .

Then the score is calculated as:

$$p_{xy}^{(k)} = \frac{(\text{deg}(y))^{-d^{(k)}}}{e^{-\text{rec}(x,y)r^{(k)}}} \cdot \sigma \left(\sum_{i \in |\mathcal{E}|} \frac{\text{ev}(e_i, x, y)}{\text{ev}(x, y)} e_i^{(k)} + b_e^{(k)} \right) \cdot \sigma(\text{ev}(x, y)e^{(k)} + b^{(k)}),$$

where $\sigma(x) = \frac{1}{1+e^{-x}}$ and $d^{(k)}$, $r^{(k)}$, $e_i^{(k)}$, $b_e^{(k)}$, $e^{(k)}$, $b^{(k)}$ are model parameters.

B. Model parameters

The total number of P3LTR model's parameters equals $3 \cdot (5 + |\mathcal{E}|)$.

The RP3Beta model is not learnable and has only two hyperparameters: α and β . Our model is equivalent to RP3Beta model when $d^{(1)} = d^{(2)} = \alpha$, $d^{(3)} = \beta$ and all other parameters equal 0.

RP3Beta is a generalization of P3Alpha [27] (for $\beta = 0$), P3 [27] (for $\alpha = 1$, $\beta = 0$) and #3-Paths [27] (for $\alpha = 0$, $\beta = 0$), so naturally P3LTR also includes these methods.

Parameters $d^{(k)}$ tell us how impactful (destination) nodes of a given degree should be, i.e., $d^{(k)} = 0$ means that we treat all nodes equally, $d^{(k)} > 0$ means that we reduce the impact of nodes with a greater degree, $d^{(k)} < 0$ means that we increase the impact of nodes with the greater degree.

Parameters $r^{(k)}$ are used to utilize the recency of interactions, i.e., when $r^{(k)} > 0$ more recent interactions have higher importance, when $r^{(k)} = 0$ the recency of interactions has no impact on recommendations and when $r^{(k)} < 0$ the older interactions are more impactful.

Parameters $e_i^{(k)}$, $b_e^{(k)}$ are designed to utilize the information about type of interactions between users and items. The parameters associated with events requiring higher user engagement (e.g., replying to an offer) might have higher values than the others (e.g., the parameter associated with visiting an offer).

Parameters $e^{(k)}$, $b^{(k)}$ were introduced to include the information about the frequency of interactions. Higher frequency is an indicator of higher user engagement and might be used to increase the importance of a particular item.

C. Model training

The goal of training the model is to learn the values of the parameters of our feature encoders $\phi^{(k)}$. We describe three

components of this process: a forward pass for a single user, a training loop, and a loss function.

1) *Forward pass for a single user*: We can present our model from the perspective of a message-passing paradigm used in graph convolutional neural networks [15], [24], [25], [26]. For the initial representation ($k = 0$) we set the score $r_u^{(0)} = 1$ for the node representing the given user and the score 0 for all other nodes. Then for all nodes x and for $k = 1, 2, 3$ we perform the message passing:

$$r_x^{(k)} = \sum_{y \in \mathcal{N}(x)} r_y^{(k-1)} \phi^{(k)}(f_x^n, f_y^n, f_{xy}^e).$$

This process can be interpreted as spreading the message across the graph. At the beginning, we send the message to the neighboring nodes depending on their relevancy calculated by $\phi^{(1)}$. Then each of these nodes sends the message to their neighbors with respect to the relevancy calculated by $\phi^{(2)}$. This process could be continued, but for efficiency reasons and based on the results of Cooper *et al.* [27], we limit it to 3 steps.

2) *Training loop*: A training process is described by Algorithm 1.

Algorithm 1 Training loop of the P3LTR model.

for iteration = 1, 2, ..., iterations **do**

 Update edge weights of the graph based on feature encoders

 Set the loss to 0.

for i = 1, 2, ..., batch size **do**

 Pick a random target user (by default: random user who interacted with at least 2 items) and take his most recent interacted item as a *validation node*.

 Make a forward pass for this user and calculate the scores of top k items and the score and position of a validation node.

 Calculate the loss for this user and add it to the current loss.

end for

 Backpropagate the loss and update the weights of feature encoders.

end for

3) *Loss function*: The idea of our loss function is to score the validation item higher than the other items. Let us define

$$\text{ratio} = \frac{\text{avg score of top k items}}{\text{validation node score}}.$$

To stabilize the training, we additionally calculated the sum of squares of the parameters and multiplied it by a constant regularization parameter. We considered three loss functions:

- **ratio**: ratio + regularization term,
- **log ratio**: $\ln(\text{ratio})$ + regularization term,
- **boosted log ratio**: $\ln(\text{ratio}) \cdot \text{validation node position} + \text{regularization term}$.

The idea of the log ratio was inspired by BPR loss [34] function and the idea of the boosted log ratio was inspired

by WARP loss [35]. The best loss function was chosen during the hyperparameter optimization.

D. Model advantages

We would like to emphasize the following advantages of the proposed approach:

- 1) P3LTR generalizes RP3Beta which is a strong baseline model.
- 2) P3LTR directly utilizes the information about the user-item relationship. For instance, our model may be used for encoding the importance of ratings in the explicit feedback dataset used for the top N recommendations task if we treat each rating as a different type of interaction.
- 3) P3LTR utilizes additional information regarding the users and the items. In our collaborative filtering dataset, we used only node degrees as such features, but we can easily extend the model to include additional user and item features.
- 4) P3LTR is an explainable model from two perspectives: we can explain because of which items a given item is recommended and explain why some items are more influential on recommendations.
- 5) P3LTR directly utilizes the information of the node's neighbors. Such an approach might give better results than embedding-based approaches for users with a low number of interactions.
- 6) The training pipeline is used only for optimizing the weights of feature encoders. Hence it can be trained sporadically (or even just once) and be utilized for providing predictions every day.
- 7) The model prediction is almost as efficient, as RP3Beta. The difference is in the preprocessing stage, where in P3LTR, we need to additionally calculate features and pass them through feature encoders.

IV. EXPERIMENTAL SETUP

A. Dataset

We utilized the OLX Jobs Interactions dataset which is publicly available on Kaggle¹. In our previous work, as yet unpublished, we compared several collaborative filtering non-neural approaches. The RP3Beta model outperformed other approaches in terms of accuracy and efficiency and, after online A/B tests, has been deployed at OLX.

The dataset contains 65 502 201 events made on <http://olx.pl/praca> by 3 295 942 users who interacted with 185 395 job ads in 2 weeks of 2020. Each event contains 4 pieces of information: user id, item id, event type (e.g., click or reply) and timestamp.

It is important to note that users usually do not interact with many job ads (average: 20, median: 6, first quartile: 2, third quartile: 18).

¹<https://www.kaggle.com/datasets/olxdatscience/olx-jobs-interactions>

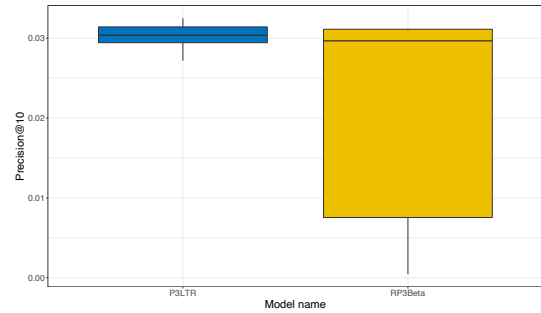


Fig. 2. Precision for each model depending on parameters.

B. Train-test splitting

We split the events into train and test sets by time, i.e., 20% of the newest events (approximately 2.8 days) were included in the test set. We filtered out from the test set all user-item pairs which appeared in the train set (to avoid recommending already seen items).

C. Hyperparameter tuning

For the sake of efficiency, we extracted 20% of users and 20% of items from the original train set and, according to the train-test splitting technique described in the previous section, we divided them into train and test sets used for validation. For each model, we defined the hyperparameter space and performed 100 iterations chosen by Bayesian optimization using Gaussian processes. We were optimizing for precision@10 [36] calculated on 30 thousand users. In Fig. 2 we can observe that the hyperparameters significantly affect the performance of tuned models. Therefore, tuning was essential for providing reliable results for compared methods. We can also see that choosing suboptimal hyperparameters for the RP3Beta model can result in very poor performance, which is not the case for P3LTR. We report the optimal hyperparameters in Table I.

V. RESULTS

We used the best found hyperparameters to train our model on the full dataset and generate recommendations for all 619 389 users in the test set. We compared the following methods:

- **P3LTR**,
- **RP3Beta**,
- **P3**: which is the RP3Beta model for $\alpha = 1$ and $\beta = 0$,
- **#3-Paths**: which is the RP3Beta model for $\alpha = 0$ and $\beta = 0$. We initialized all the parameters of our P3LTR model to zeros, which makes the #3-Paths model equivalent to the P3LTR model before the learning process.

In this section, we compare the accuracy and diversity of these models. We will also discuss the parameters of our P3LTR model.

A. Accuracy

In Table II we list common accuracy evaluation metrics calculated with respect to the top 10 recommendations. The

TABLE I
MODEL HYPERPARAMETERS.

Model	Model hyperparameters
RP3beta	{'alpha': 0.61447198, 'beta': 0.1443548}
P3LTR	{'regularization': 0.001, 'learning_rate': 0.02, 'batch_size': 153, 'iterations': 80, 'top_k': 205, 'loss': 'log_ratio'}

TABLE II
ACCURACY RESULTS. ALL PRESENTED METRICS WERE DESCRIBED IN [36].

Model	P3LTR	RP3Beta	P3	#3-Paths
precision	0.0515	0.0484	0.0481	0.0391
recall	0.0817	0.0783	0.0782	0.0611
ndcg	0.0798	0.0759	0.0755	0.0599
mAP	0.0414	0.0393	0.0390	0.0302
MRR	0.1423	0.1365	0.1363	0.1107
LAUC	0.5408	0.5391	0.5391	0.5305
HR	0.3242	0.3131	0.3147	0.2605

values should not be directly compared with the results achieved on other datasets, because metrics heavily depend on the distribution of the dataset (for example high sparsity) and the train/test splitting strategy. We can observe that our method P3LTR outperforms RP3Beta with respect to all listed metrics.

To identify differences between the methods, we test the null hypothesis that all methods perform the same. We used the Friedman test with Iman and Davenport extension. The p -value from this test is equal to 0 which indicates that we can safely reject the null hypothesis that all the algorithms perform the same. We can therefore proceed with the post-hoc tests in order to detect significant differences among all of the methods. Demšar [37] proposes the use of the Nemenyi's test and preparing a plot to visually check the differences, the critical difference plot. In the plot, those algorithms that are not joined by a line can be regarded as different. In our case, with a significance of $\alpha = 0.05$ any two algorithms with a difference in the mean rank above 0.006 are regarded as non-equal (Fig. 3).

We can observe three disjoint groups of methods:

- 1) P3LTR,
- 2) RP3Beta and P3,
- 3) #3-Paths.

From this analysis, we see that P3LTR performs significantly better than other methods on the examined dataset.

B. Diversity

Most of the job ads refer to only one job position. Hence we should avoid recommending the same item to a great number of users. For that reason, we report also the diversity metrics in Table III. Test coverage is a fraction of items from the test set which were recommended to at least one user. We also report Shannon entropy [38] and Gini index [38]. We can see that P3LTR is the most diverse method with respect to all these metrics. We can also note that the #3-Paths method seems less

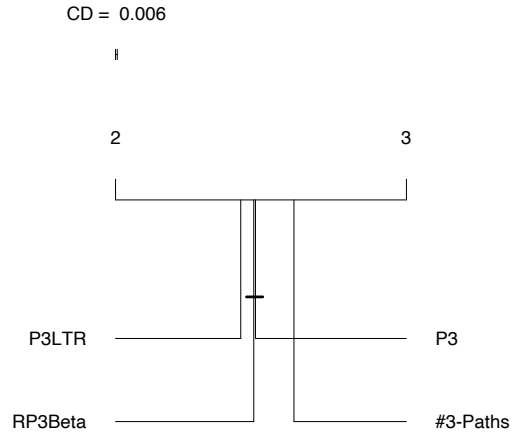


Fig. 3. Critical difference plot (for precision).

TABLE III
DIVERSITY RESULTS.

Model	P3LTR	RP3Beta	P3	#3-Paths
test coverage	0.757	0.573	0.617	0.374
Shannon entropy	10.090	9.527	9.631	8.712
Gini index	0.844	0.908	0.898	0.957

diverse than other methods. We suppose that the reason is that this method recommends the items based on the number of paths of length 3 connecting a given user and item, so the most popular items are more often recommended.

In order to decide whether to deploy a new recommendation system in production, we usually check how different are the recommendations produced by a new model compared to the old one. To assess it we calculated the overlap coefficient [39] with respect to user-item pairs. The results are reported in Table IV. We see that 70% of the top 10 recommendations provided by P3LTR and RP3Beta models are the same recommendations. We can also observe that RP3Beta and P3 provide pretty similar results on our dataset (overlap coefficient equals 84%).

C. Parameters of the P3LTR model

As we mentioned, the parameters of our model can be easily interpreted. In the Table V we report the values of $d^{(k)}$ (parameters related to node degrees) and $r^{(k)}$ (parameters related to recency).

In previous works regarding the RP3Beta model, usually positive values for α and β were chosen to discourage the model from recommending the most popular items [1], [3].

TABLE IV
AN OVERLAP OF MODELS.

Model	P3LTR	RP3Beta	P3	#3-Paths
P3LTR	100%	70%	64%	50%
RP3Beta	70%	100%	84%	68%
P3	64%	84%	100%	60%
#3-Paths	50%	68%	60%	100%

TABLE V
PARAMETERS OF THE P3LTR MODEL.

Parameter	$k = 1$	$k = 2$	$k = 3$
$d^{(k)}$	0.631	0.163	0.433
$r^{(k)}$	0.081	0.008	0.028

We can see that in our machine learning approach also positive values were learned for all $d^{(k)}$ parameters.

Additionally, in RP3Beta model we have $d^{(1)} = d^{(2)} = \alpha$. However, P3LTR model chose very different values for $d^{(1)}$ and $d^{(2)}$.

Regarding recency, the model chose positive values of $r^{(k)}$ parameters. It means that the more recent interactions should have higher importance.

We do not discuss parameters related to event type e.g., viewing or replying to an ad, because they did not converge within the number of iterations we have chosen. Hence the reported results might differ when we train the model multiple times. We believe the convergence could be achieved with a greater value of a batch_size hyperparameter, but it would also significantly increase the training time.

VI. SUMMARY

In the paper, we introduced a new graph vertex ranking recommendation method which we named P3LTR. It generalizes the RP3Beta model which provides very efficient and accurate recommendations on multiple datasets. We described several strengths of our approach, including explainability and prediction efficiency. We showed that our method is superior to RP3Beta on the OLX Jobs Interactions dataset in terms of accuracy and diversity of recommendations.

The proposed method may improve the quality of recommendations currently being generated using the RP3Beta model that is implemented at OLX Jobs in a production setting.

In future work, we plan to explore more advanced feature encoders which utilize user and item features. We would like to explore and compare different loss functions for the P3LTR model. Additionally, we would like to launch A/B tests on production to measure the model's effectiveness on real users.

REFERENCES

- [1] B. Paudel, F. Christoffel, C. Newell, and A. Bernstein, "Updatable, accurate, diverse, and scalable recommendations for interactive applications," *ACM Transactions on Interactive Intelligent Systems*, vol. 7, pp. 1–34, 12 2016.
- [2] M. F. Dacrema, P. Cremonesi, and D. Jannach, "Are we really making much progress? a worrying analysis of recent neural recommendation approaches," in *Proceedings of the 13th ACM Conference on Recommender Systems*, ser. RecSys '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 101–109.
- [3] M. F. Dacrema, S. Boglio, P. Cremonesi, and D. Jannach, "A troubling analysis of reproducibility and progress in recommender systems research," *ACM Transactions on Information Systems*, vol. 39, pp. 1–49, 01 2021.
- [4] V. W. Anelli, A. Bellogín, T. Di Noia, and C. Pomo, "Reenvisioning the comparison between neural collaborative filtering and matrix factorization," in *Fifteenth ACM Conference on Recommender Systems*, ser. RecSys '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 521–529.
- [5] C. Gómez-Urbe and N. Hunt, "The netflix recommender system," *ACM Transactions on Management Information Systems*, vol. 6, pp. 1–19, 12 2015.
- [6] B. Smith and G. Linden, "Two decades of recommender systems at amazon.com," *IEEE Internet Computing*, vol. 21, pp. 12–18, 05 2017.
- [7] M. J. Pazzani and D. Billsus, "Content-based recommendation systems," in *The Adaptive Web*. Springer, 2007.
- [8] U. Javed, K. Shaukat Dar, I. Hameed, F. Iqbal, T. Mahboob Alam, and S. Luo, "A review of content-based and context-based recommendation systems," *International Journal of Emerging Technologies in Learning (IJET)*, vol. 16, 02 2021.
- [9] R. Chen, K. Hua, Y.-S. Chang, B. Wang, L. Zhang, and X. Kong, "A survey of collaborative filtering-based recommender systems: From traditional methods to hybrid methods based on social networks," *IEEE Access*, vol. PP, pp. 1–1, 10 2018.
- [10] X. Wang, X. He, M. Wang, F. Feng, and T.-S. Chua, "Neural graph collaborative filtering," in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR'19. New York, NY, USA: Association for Computing Machinery, 2019, p. 165–174. [Online]. Available: <https://doi.org/10.1145/3331184.3331267>
- [11] X. Ning and G. Karypis, "Slim: Sparse linear methods for top-n recommender systems," in *Proceedings of the 2011 IEEE 11th International Conference on Data Mining*, ser. ICDM '11. USA: IEEE Computer Society, 2011, p. 497–506. [Online]. Available: <https://doi.org/10.1109/ICDM.2011.134>
- [12] H. Khojamli and J. Razmara, "Survey of similarity functions on neighborhood-based collaborative filtering," *Expert Systems with Applications*, vol. 185, p. 115482, 2021.
- [13] M. Kula, "Metadata embeddings for user and item cold-start recommendations," *arXiv preprint arXiv:1507.08439*, 2015.
- [14] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative filtering for implicit feedback datasets," in *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, ser. ICDM '08. USA: IEEE Computer Society, 2008, p. 263–272.
- [15] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, and M. Wang, *LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation*. New York, NY, USA: Association for Computing Machinery, 2020, p. 639–648. [Online]. Available: <https://doi.org/10.1145/3397271.3401063>
- [16] M. Grbovic, V. Radosavljevic, N. Djuric, N. Bhamidipati, J. Savla, V. Bhagwan, and D. Sharp, "E-commerce in your inbox: Product recommendations at scale," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '15. New York, NY, USA: Association for Computing Machinery, 2015, p. 1809–1818.
- [17] O. Barkan and N. Koenigstein, "Item2vec: Neural item embedding for collaborative filtering," 09 2016, pp. 1–6.
- [18] G. Adomavicius and A. Tuzhilin, *Context-Aware Recommender Systems*. Boston, MA: Springer US, 2015, pp. 191–226. [Online]. Available: https://doi.org/10.1007/978-1-4899-7637-6_6
- [19] S. Kulkarni and S. F. Rodd, "Context aware recommendation systems: A review of the state of the art techniques," *Computer Science Review*, vol. 37, p. 100255, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1574013719301406>
- [20] M. Quadrana, P. Cremonesi, and D. Jannach, "Sequence-aware recommender systems," *ACM Comput. Surv.*, vol. 51, no. 4, jul 2018. [Online]. Available: <https://doi.org/10.1145/3190616>
- [21] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," vol. 2016, 07 2016, pp. 855–864.

- [22] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 03 2014.
- [23] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, ser. NIPS'13. Red Hook, NY, USA: Curran Associates Inc., 2013, p. 3111–3119.
- [24] X. Wang, X. He, M. Wang, F. Feng, and T.-S. Chua, "Neural graph collaborative filtering," ser. SIGIR'19. New York, NY, USA: Association for Computing Machinery, 2019, p. 165–174. [Online]. Available: <https://doi.org/10.1145/3331184.3331267>
- [25] R. van den Berg, T. Kipf, and M. Welling, "Graph convolutional matrix completion," *ArXiv*, vol. abs/1706.02263, 2017.
- [26] R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, and J. Leskovec, "Graph convolutional neural networks for web-scale recommender systems," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 974–983. [Online]. Available: <https://doi.org/10.1145/3219819.3219890>
- [27] C. Cooper, S. H. Lee, T. Radzik, and Y. Siantos, "Random walks in recommender systems: Exact computation and simulations," in *Proceedings of the 23rd International Conference on World Wide Web*, ser. WWW '14 Companion. New York, NY, USA: Association for Computing Machinery, 2014, p. 811–816.
- [28] F. Fouss, A. Pirotte, and M. Saerens, "A novel way of computing similarities between nodes of a graph, with application to collaborative recommendation," in *The 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI'05)*, 2005, pp. 550–556.
- [29] F. Fouss, A. Pirotte, J.-m. Renders, and M. Saerens, "Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 3, pp. 355–369, 2007.
- [30] M. Gori and A. Pucci, "Itemrank: A random-walk based scoring algorithm for recommender engines," 01 2007, pp. 2766–2771.
- [31] X. Geng, H. Zhang, J. Bian, and T.-S. Chua, "Learning image and user features for recommendation in social networks," in *2015 IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 4274–4282.
- [32] C. Wang and D. Blei, "Collaborative topic modeling for recommending scientific articles," 08 2011, pp. 448–456.
- [33] F. M. Harper and J. A. Konstan, "The movielens datasets: History and context," *ACM Trans. Interact. Intell. Syst.*, vol. 5, no. 4, dec 2015. [Online]. Available: <https://doi.org/10.1145/2827872>
- [34] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "Bpr: Bayesian personalized ranking from implicit feedback," *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence, UAI 2009*, 05 2012.
- [35] J. Weston, S. Bengio, and N. Usunier, "Wsabie: Scaling up to large vocabulary image annotation," 01 2011, pp. 2764–2770.
- [36] Y.-M. Tamm, R. Damdinov, and A. Vasilev, "Quality metrics in recommender systems: Do we calculate metrics consistently?" in *Fifteenth ACM Conference on Recommender Systems*, ser. RecSys '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 708–713. [Online]. Available: <https://doi.org/10.1145/3460231.3478848>
- [37] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.
- [38] G. Shani and A. Gunawardana, *Evaluating Recommendation Systems*. Boston, MA: Springer US, 2011, pp. 257–297. [Online]. Available: https://doi.org/10.1007/978-0-387-85820-3_8
- [39] M. Vijaymeena and K. Kavitha, "A survey on similarity measures in text mining," *Machine Learning and Applications: An International Journal*, vol. 3, pp. 19–28, 03 2016.

Personality Prediction from Social Media Posts using Text Embedding and Statistical Features

Seiyu Majima and Konstantin Markov
 University of Aizu
 Aizuwakamatsu, Fukushima, Japan
 Email: m5241125, markov@u-aizu.ac.jp

Abstract—Recent advances in deep learning based language models have boosted the performance in many downstream tasks such as sentiment analysis, text summarization, question answering, etc. Personality prediction from text is a relatively new task that has attracted researchers’ attention due to the increased interest in personalized services as well as the availability of social media data. In this study, we propose a personality prediction system where text embeddings from large language models such as BERT are combined with multiple statistical features extracted from the input text. For the combination, we use the self-attention mechanism which is a popular choice when several information sources need to be merged together. Our experiments with the Kaggle dataset for MBTI clearly show that adding text statistical features improves the system performance relative to using only BERT embeddings. We also analyze the influence of the personality type words on the overall results.

I. INTRODUCTION

PERSONALITY research has a long history of studies mainly in psychology where stable patterns of thoughts, feelings, and behaviors have been associated with the so called personality traits. They are useful indicators for describing individual’s preferences in perceiving the world and making decisions [1].

Recent advances in natural language processing have made it possible to build machine learning models using online data on human behavior and preferences to automatically predict people’s personality traits. Applications include wide variety of internet services including recommender systems [2], product personalization [3], social network and sentiment analysis [4], [5].

In psychological science, there are two widely adopted models for formal description of the personality traits. The five factor model (Big Five) [6] consists of five broad dimensions of personality - Openness, Conscientiousness, Agreeableness, Extraversion, and Neuroticism. Individual’s scores on each of these dimensions is obtained using a standardized self-report questionnaires. In the other model, personality is formally described by 16 types known as MBTI (Myers-Briggs Type Indicator) [7]. MBTI is an introspection self-reported diagnostic test aimed at showing psychological preferences about how individuals perceive the world and make decisions. The subjects are classified into 16 personality types that are created from the combination of binary assignments to four dimensions: Introversion versus Extraversion (I/E), Sensing versus Intuiting (S/I), Thinking versus Feeling (T/F), and



Fig. 1. MBTI type personality keys [7]

Judging versus Perceiving (J/P) as shown in Fig. 1. It is been long considered that personality is reflected in individual’s use of language [8]. People with high score in extraversion use more positive emotion words while those higher in neuroticism favor first-person words such as “I”, “my”, and “me”.

A significant number of studies have been dedicated to automatic personality prediction. As an input modality, text data are widely used because they are easy to collect, though video has also been used lately [9]. Some of the first works have focused on text features based on lexicon, syntax, etc., and investigated their correlation with the personality traits as well as their classification performance using shallow machine learning models [10], [11], [12], [13]. Others rely on the commonly used TF-IDF features, for example [14], where personality traits are predicted using an XGBoost classifier.

Some recent works utilize the achievements in the neural networks based text processing by using word embeddings from pre-trained Word2Vec [15] or GloVe [16] models as well as big language models such as BERT [17], [18]. In [17], BERT embeddings are compared with a set of psycholinguistic features and has been found that the BERT derived features perform better.

In this study, as a starting point we also use BERT derived embeddings like in [17], but then we try to combine them with a set of different statistical features extracted from the input text documents. Those features include uni-gram and bi-gram histograms, topic distribution, post and word length statistics,

etc. In order to combine them with the BERT document embeddings in an efficient way, we use a self-attention based method.

II. SYSTEM DESCRIPTION

We assume that the text data consist of multiple posts from various users with known MBTI labels. The system takes all posts from a single user as input data and outputs four dimensional binary vector where each element corresponds to one of the four MBTI axes, i.e. E/I, N/S, T/F and P/J. For example, if the output is $[1, 0, 0, 1]$, the personality type is ESFP. Since there are only 16 possible personality types, we cast the personality prediction task as a classification task with 16 categories which are then projected onto the four MBTI axes.

Input text data from each user are transformed into a document vector using pre-trained BERT language model. In our base system, document vectors from all users are passed to a simple MLP classifier with 16 outputs. In the full system, in addition to the document vector, several statistical features are extracted from the input text data and linearly transformed to match the document vector's dimension. Then, using self-attention mechanism, all vectors are combined into a single final vector which is passed to the the same MLP classifier.

A. Text Embedding

In natural language processing applications it has become popular to adopt the transfer learning approach where pre-trained language models build from large amounts of data, such as BERT or GPT, are fine tuned or used to extract features from text for further processing by smaller machine learning models.

In this study, we selected the BERT-large model since it provides 1024-dimensional vector representation, i.e. embedding, for each input word token. In order to obtain a single vector for the whole document, we aggregate the BERT outputs by taking their average as was proposed in [17]. Fig. 2 shows the document vector extraction procedure.

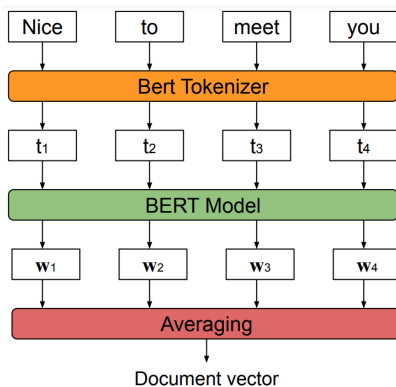


Fig. 2. Document vector extraction procedure.

Another popular way of obtaining representation vector for the whole input is to use the BERT output for the [CLS] token

[19]. This, however, works when the number of input tokens is less than maximum input length of the language model which was not the case for the majority of the users data in our experiments.

After document vectors for all the users are obtained, we use a simple MLP network to classify them into one of the 16 MBTI categories. The winning category is then transformed into four dimensional MBTI axes vector. This is our base system and its block diagram is shown in Fig. 3. It is similar to the system investigated in [17], but the main difference is the way we aggregate the BERT outputs.

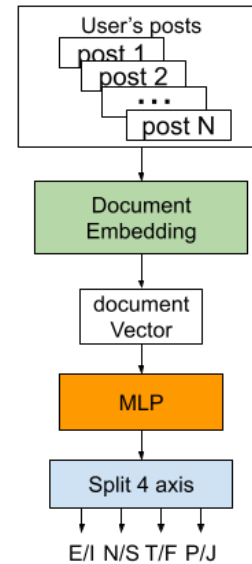


Fig. 3. Base system using only text embedding.

B. Statistical Features

Although the BERT is a very powerful language model, it is designed to capture and learn dependencies mainly at word and sentence level. When the task is to extract information at a higher user level, as in our case, some useful user dependent characteristics of the input text may get "overlooked" by the BERT model. For example, the usage of some specific words or phrases, or special symbols like emojis is difficult to obtain from the language model. However, it is easy to obtain such information using statistical analysis of users text.

1) *Uni-gram and Bi-gram Histogram*: Different people tend to have their own vocabulary of most used words and phrases and we suppose that the personality plays some role in the formation of this vocabulary. In order to get user specific vocabulary representation we use a histogram of uni-grams (single words) and bi-grams (pairs of words) present in the use data.

First, we create a global vocabulary from all users text data. Then, for each user, the frequency of each uni-gram or bi-gram is obtained from the user's data forming a histogram feature vector. Stop words like "I", "and", "the" which are common and do not provide any discriminating information

are removed from the vocabulary. Rare words, i.e. words with frequency less than a specified threshold, are removed as well.

2) *Topic Distribution*: Another factor that may be influenced by the user’s personality is the topic of the user’s post. There various ways to determine the topic of a given text. Topic classification into pre-determined categories such as news, politics, sports, etc. has been studied for years [20]. In our case, however, we are more interested in the topic differences among the posts rather than their labels. Furthermore, the granularity level of the topic categories may result in quite different classification results. That is why we adopted an unsupervised topic learning approach.

First, each post from all users is transformed into a vector using the same approach as in our base system, i.e. using BERT language model. Post vectors obtained this way are clustered into several clusters with the K-means algorithm. Then for each user, cluster occupancy histogram of its posts is used as a topic distribution feature vector.

3) *Post and Word Length Statistics*: The number of words in a post can vary significantly depending on various factors one of which we assume is the personality type. If there is any correlation, it can be revealed by taking the first and second order statistics of the word number in a post. Extending this idea to the word length in letters, for each user we construct a feature vector from the mean and variance of word number per post and letter number per word.

4) *Emoticon Usage*: It’s a common practice to use emoticons in users posts to express emotions. Some examples of most often used emoticons are given in Table I. We suppose that different people may use different sets of emoticons and the frequency of their usage may be related to personality types. In fact, in [21], emoji embeddings have been concatenated with word embeddings in an attention-based BiLSTM model.

Based on the set of emoticons found in the dataset, for each user we obtain a histogram of emoticons present in its posts which is used as a feature vector.

TABLE I
SOME FREQUENTLY USED EMOTICONS AND THEIR MEANING

Emoticon	Meaning
:)	happy face
:D	laughing
:’(crying
:/	annoyed

C. Self-Attention based Embedding and Feature Combination

As we described in Section II-A, our base system uses text embedding to predict user’s personality type. By combining the text embedding with the feature vectors obtained from the statistical text analysis we aim at improving the system performance.

There are various way to combine vectors representing different information sources including simple concatenation, weighted sum, etc. In this study, we adopt the weighted sum approach where weights are calculated dynamically based

on the current input and a learned stream importance. This approach is implemented using a self-attention network [22].

Given vectors v_1, v_2, \dots, v_N , we first, pass them through a vector specific linear layer with sigmoid activation function

$$u_i = \text{sig}(W_i v_i + b_i) \quad (1)$$

The combined output vector o is then calculated as a weighted sum of vectors v_i as follows

$$o = \sum_i a_i v_i \quad (2)$$

where weights a_i are obtained using softmax function

$$a_i = \frac{\exp u_i}{\sum_j \exp u_j} \quad (3)$$

We have to note that the original text embedding vector and the statistical feature vectors have different sizes. In order to equalize their dimensionality, each statistical feature vector is passed through a linear layer with a proper weight matrix and no bias.

Fig. 4 shows the block diagram of our system where the output of the self-attention network o is used as input to the personality prediction MLP classifier.

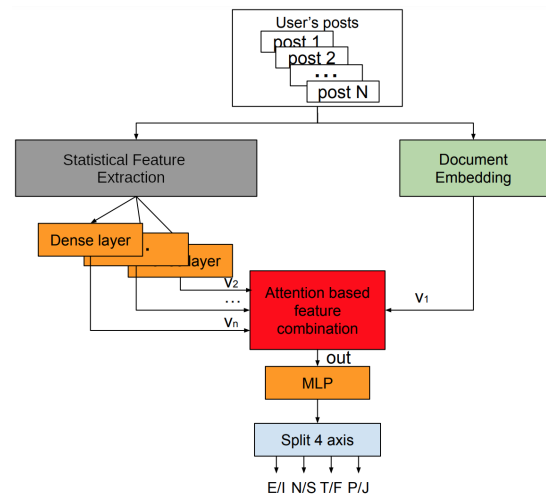


Fig. 4. Self-attention based text embedding and statistical features combination system.

III. EXPERIMENTS

A. Dataset

In this study we use the Kaggle MBTI personality type dataset [23]. The data were collected through the Personality-Cafe forum and include diverse selection of posts from people interacting in an informal online social setting.

This dataset contains records of the last 50 posts from 8600 PersonalityCafe users along with their MBTI binary personality type. The MBTI label distribution is given in Table II which shows that the data are quite unbalanced.

TABLE II
DISTRIBUTION OF THE MBTI TYPE LABELS.

MBTI type label	Number of labels
E/I	1999/6676
N/S	1197/7478
T/F	4694/3981
P/J	5241/3434

B. Data Pre-processing

First, all text data were cleaned which is a standard text pre-processing practice. This includes case conversion, reducing repeated characters and punctuations, expanding contractions, removing numbers, etc. There was a substantial number of URL links which we replaced with a special token [URL]. Posts consisting of URLs only were removed.

C. Model Training

For model training and evaluation we adopted a 10-fold cross-validation scheme shown in Fig. 5. In each fold, 10% of the data is reserved for testing. The rest is divided into training and validation sets with 9:1 proportion. This way, we train 10 different models and tune their hyper-parameters on the corresponding validation set. After that, each model is evaluated on the fold's test data and the results are averaged over the folds.

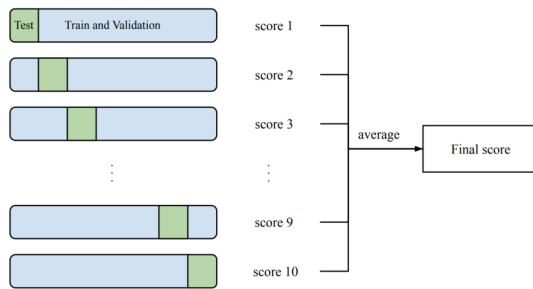


Fig. 5. 10-fold cross-validation scheme for model training and evaluation.

D. Evaluation Metrics

For classification tasks, the standard evaluation metrics are *Accuracy* and/or *F1-score* and we are reporting all the results in term of those two metrics.

In order to be able to compare our results with those already published, we calculate the *Accuracy* and *F1-score* separately for each of the four MBTI axes, i.e. E/I, N/S, T/F, P/J and for the overall system performance we take their average.

E. Results

How efficient would be a combination of several feature vectors depends not only on the way they are combined, but to a large extent to how much discriminating information each feature contributes. It is difficult to assess this contribution quantitatively, so we analysed the differences in feature vectors representing different users. For example, in the uni-gram case, these are the word usage histograms. A comparison of such histograms for two users is shown in Fig. 6. It is apparent

that they are quite different and could be helpful for the user discrimination.

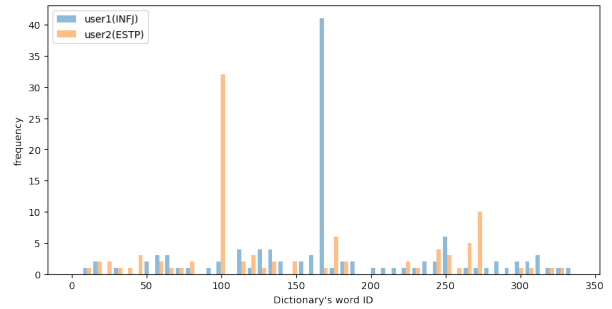


Fig. 6. Uni-gram histogram of two users.

For the topic distribution feature, we needed to tune the number of topic clusters used to create it. This number is a hyper-parameter which can be determined manually since the range of possible values is not that wide. We created topic models with 2 to 16 clusters and evaluated them by combining the topic feature vectors with document vectors from BERT. The obtained average Accuracy and F1-score are given in Fig. 7. It is clear that the optimum number of topic clusters is 4.



Fig. 7. Classification performance with respect of the number of topic clusters.

The performances of our base system, i.e. using only document embedding as feature vector, and the proposed system with self-attention based combination with statistical features are summarized in Table III and Table IV. The first row in each table shows the base system results. The following rows show the result when the document embedding vectors are combined with one of the uni-gram (uni), bi-gram (bi), topic distribution (t), post and word length (p), and emoticon usage (e) features. The last row shows the performance of the best multiple statistical features combination.

As can be seen from these tables, each additional statistical feature slightly improved the base system results. Uni-gram and bi-gram features scored almost the same which may be explained by the limited vocabulary and the amount of training data per user. The highest result, however, was obtained with multiple features combination.

Finally, in Fig. 8 we compare our best result with the results from some other studies where the same Kaggle MBTI dataset

TABLE III
ACCURACY (%) OF THE TEXT EMBEDDING (BASE) AND ITS COMBINATION WITH THE STATISTICAL FEATURE VECTORS.

Features	E/I	N/S	T/F	P/J	Ave
base	83.1	88.5	84.1	78.0	83.4
base+uni	84.4	88.7	84.5	78.5	84.0
base+bi	84.4	88.7	84.3	78.4	84.0
base+t	84.1	88.5	84.5	77.5	83.7
base+p	84.4	88.7	84.5	78.0	83.9
base+e	84.1	88.6	84.3	77.9	83.7
base+uni+p+e	84.6	88.8	84.5	78.7	84.2

TABLE IV
F1-SCORE OF THE TEXT EMBEDDING (BASE) AND ITS COMBINATION WITH THE STATISTICAL FEATURE VECTORS.

Features	E/I	N/S	T/F	P/J	Ave
base	0.753	0.664	0.839	0.764	0.755
base+uni	0.758	0.691	0.844	0.772	0.767
base+bi	0.761	0.689	0.841	0.771	0.766
base+t	0.756	0.685	0.841	0.761	0.761
base+p	0.762	0.693	0.844	0.766	0.766
base+e	0.758	0.690	0.841	0.765	0.763
base+uni+p+e	0.762	0.695	0.844	0.773	0.769

was used. In [14], the TF-IDF features are used with XGBoost classifier while in [17] BERT model is combined with an MLP network.

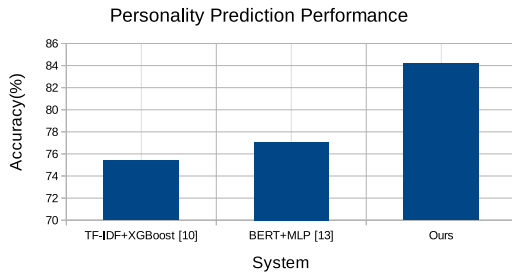


Fig. 8. Comparison of our and other published systems.

IV. DISCUSSION

It is well known that "models are as good as the data" they are trained on. The data samples and truth labels quality plays essential role in the final systems performance. This is an important issue especially when the data are collected from social media.

During the analysis of the Kaggle MBTI dataset we noticed that all users refer to their own MBTI type more often than to the other personality types. Fig. 9 shows a heatmap of the frequency of the MBTI type word usage by each personality type users. It is clear that user's own MBTI type word is mentioned several times more frequently in their posts. This suggests that a histogram vector of personality type words usage can be highly discriminative for this dataset.

Thus, as an additional experiment, we trained an MLP classifier with only MBTI histogram feature vectors (one per user) as well as their combination with document embedding vectors. The results shown in Table V were surprising, though

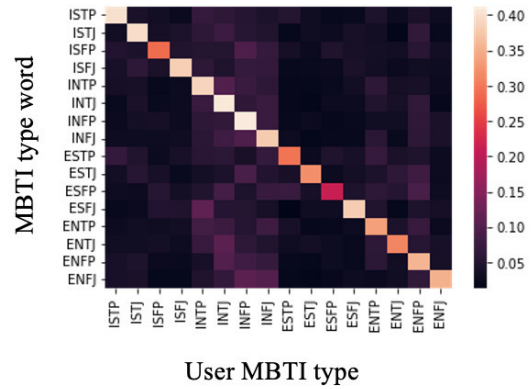


Fig. 9. Heatmap of MBTI type word usage by users of different personality types.

not unexpected. The MBTI feature alone outperformed our

TABLE V
ACCURACY (%) OF THE BASE SYSTEM, MBTI FEATURES AND THEIR COMBINATION.

Features	E/I	N/S	T/F	P/J	Ave
base	83.1	88.5	84.1	78.0	83.4
mbti	86.2	90.5	84.8	80.3	85.4
base+mbti	87.2	91.3	86.1	81.8	86.6

best system and when combined with the BERT document features improved the result by almost 4%. Of course, this outcome is specific to the Kaggle dataset and will not hold with other data collections. Nevertheless, it underlines the importance of the data when building machine learning systems.

V. CONCLUSION

In this paper, we proposed a personality type prediction system which combines text document embedding vectors obtained from the BERT language model with statistical feature vectors extracted from the text data. Using powerful language models for downstream tasks has been proved quite effective and our results confirm this conclusion. However, there is always room for improvement when additional task specific knowledge is incorporated in the system as in our usage of statistical text information.

System evaluation with a single dataset reveals only the potential effect and efficiency of the proposed approach and further experimentation with other data collections is necessary in order to prove its merits.

REFERENCES

- [1] S. M. Sarsam, H. Al-Samarraie, and A. I. Alzahrani, "Influence of personality traits on users' viewing behaviour," *Journal of Information Science*, April 2021. [Online]. Available: <https://doi.org/10.1177/0165551521998051>
- [2] S. C. Matz, M. Kosinski, G. Nave, and D. J. Stillwell, "Psychological targeting as an effective approach to digital mass persuasion," *Proceedings of the national academy of sciences*, vol. 114, no. 48, pp. 12 714–12 719, 2017.

- [3] S. T. Völkel, R. Schödel, D. Buschek, C. Stachl, Q. Au, B. Bischl, M. Bühner, and H. Hussmann, "Opportunities and challenges of utilizing personality traits for personalization in hci," *Personalized Human-Computer Interaction*, vol. 31, 2019.
- [4] J. M. Balmaceda, S. Schiaffino, and D. Godoy, "How do personality traits affect communication among users in online social networks?" *Online Information Review*, 2014.
- [5] L. Yue, W. Chen, X. Li, W. Zuo, and M. Yin, "A survey of sentiment analysis in social media," *Knowledge and Information Systems*, vol. 60, no. 2, pp. 617–663, 2019.
- [6] O. P. John, L. P. Naumann, and C. J. Soto, "Paradigm shift to the integrative big five trait taxonomy: History, measurement, and conceptual issues." 2008.
- [7] P. D. Tieger, B. Barron, and K. Tieger, *Do what you are: Discover the perfect career for you through the secrets of personality type*. Hachette UK, 2014.
- [8] H. A. Schwartz, J. C. Eichstaedt, M. L. Kern, L. Dziurzynski, S. M. Ramones, M. Agrawal, A. Shah, M. Kosinski, D. Stillwell, M. E. Seligman *et al.*, "Personality, gender, and age in the language of social media: The open-vocabulary approach," *PloS one*, vol. 8, no. 9, p. e73791, 2013.
- [9] C. Suman, S. Saha, A. Gupta, S. K. Pandey, and P. Bhattacharyya, "A multi-modal personality prediction system," *Knowledge-Based Systems*, vol. 236, p. 107715, 2022.
- [10] F. Mairesse, M. A. Walker, M. R. Mehl, and R. K. Moore, "Using linguistic cues for the automatic recognition of personality in conversation and text," *Journal of artificial intelligence research*, vol. 30, pp. 457–500, 2007.
- [11] G. Carducci, G. Rizzo, D. Monti, E. Palumbo, and M. Morisio, "Twitpersonality: Computing personality traits from tweets using word embeddings and supervised learning," *Information*, vol. 9, no. 5, p. 127, 2018.
- [12] K. A. Nisha, U. Kulsum, S. Rahman, M. Hossain, P. Chakraborty, T. Choudhury *et al.*, "A comparative analysis of machine learning approaches in personality prediction using mbti," in *Computational Intelligence in Pattern Recognition*. Springer, 2022, pp. 13–23.
- [13] A. Al Marouf, M. K. Hasan, and H. Mahmud, "Comparative analysis of feature selection algorithms for computational personality prediction from social media," *IEEE Transactions on Computational Social Systems*, vol. 7, no. 3, pp. 587–599, 2020.
- [14] M. H. Amirhosseini and H. Kazemian, "Machine learning approach to personality type prediction based on the myers-briggs type indicator®," *Multimodal Technologies and Interaction*, vol. 4, no. 1, 2020. [Online]. Available: <https://www.mdpi.com/2414-4088/4/1/9>
- [15] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.
- [16] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.
- [17] Y. Mehta, S. Fatehi, A. Kazameini, C. Stachl, E. Cambria, and S. Eetemadi, "Bottom-up and top-down: Predicting personality with psycholinguistic and language model features," in *2020 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2020, pp. 1184–1189.
- [18] H. Jun, L. Peng, J. Changhui, L. Pengzheng, W. Shenke, and Z. Kejia, "Personality classification based on bert model," in *2021 IEEE International Conference on Emergency Science and Information Technology (ICESIT)*. IEEE, 2021, pp. 150–152.
- [19] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *CoRR*, vol. abs/1810.04805, 2018. [Online]. Available: <http://arxiv.org/abs/1810.04805>
- [20] L. Xia, D. Luo, C. Zhang, and Z. Wu, "A survey of topic models in text classification," in *2019 2nd International Conference on Artificial Intelligence and Big Data (ICAIBD)*. IEEE, 2019, pp. 244–250.
- [21] L. Zhou, Z. Zhang, L. Zhao, and P. Yang, "Attention-based lstm models for personality recognition from user-generated content," *Information Sciences*, vol. 596, pp. 460–471, 2022.
- [22] B. Škrlić, S. Džeroski, N. Lavrač, and M. Petkovič, "Feature importance estimation with self-attention networks," *arXiv preprint arXiv:2002.04464*, 2020.
- [23] M. J., "(MBTI) myers-briggs personality type dataset," 2017. [Online]. Available: <https://www.kaggle.com/datasnaek/mbti-type>

4th International Symposium on Rough Sets: Theory and Applications

THE RSTA symposium is devoted to the state-of-the-art and future perspectives of rough sets considered from both a theoretical standpoint and real-world applications. Rough set theory is a highly developed discipline with a large number of mechanisms potentially useful in intelligent data analysis. In our special session, we aim to invite deep research papers that address the practical problem of modeling artificial intelligence processes using techniques derived from rough sets. We also encourage scientists from other research fields to participate to initiate discussions and collaborations on other methods of data exploration and approximate computations.

TOPICS

The list of topics includes, but is not limited to:

- Artificial Intelligence
- Algebraic Logic
- Logics from Rough Sets
- Approximate Reasoning
- Clustering and Rough Sets
- Dominance-based rough sets
- Distributed Cognition
- Granular Computing
- Rough mereology
- Near sets and Proximity
- Fuzzy-rough hybrid methods
- Hybrid techniques
- Rough neural computing
- Evolutionary computation and rough set
- Image and Video Processing
- Rough Sets in Education Research
- Knowledge discovery from Databases
- Missing values
- Big data
- Bio-informatics
- Three-Way Decision Making
- Data security
- Intelligent Robotics
- Knowledge engineering
- Multimedia applications
- AHP and Decision Making
- Assistive technology and adaptive sensing systems

TECHNICAL SESSION CHAIRS

- **Artiemjew, Piotr**, Faculty of Mathematics and Computer Science, University of Warmia and Mazury in Olsztyn, Poland
- **Chelly Dagdia, Zaineb**, UVSQ, Paris-Saclay, France
- **Mani, A.**, Machine Intelligence Unit, Indian Statistical Institute, Kolkata, India

PROGRAM COMMITTEE

- **Bhattacharya, Malay**, Indian Statistical Institute, India
- **Chakraborty, Debarati**, Indian Statistical Institute, India
- **Chiru, Costin-Gabriel**, Politehnica University of Bucharest, Romania
- **Ciucci, Davide**, Università di Milano-Bicocca, Italy
- **Das, Monidipa**, Nanyang Technological University, Singapore
- **Dutta, Soma**, University of Warmia and Mazury in Olsztyn, Poland
- **Düntsche, Ivo**, Brock University, Canada
- **Gomolinska, Anna**, University of Bialystok, Poland
- **Henry, Christopher**, University of Winnipeg, Canada
- **Jana, Purbita**, Indian Institute of Technology, India
- **Janicki, Ryszard**, McMaster University, Canada
- **Li, Tianrui**, Southwest Jiaotong University, China
- **Lin, Zhe**, Department of Philosophy Xiamen University Xiamen, China
- **Ma, Minghui**, Sun Yat-Sen University, China
- **Matson, Eric**, Purdue University, USA
- **Matwin, Stan**, Dalhousie University, Canada
- **Mihálydeák, Tamás**, University of Debrecen, Hungary
- **Palmigiano, Alessandra**, The Vrije Universiteit Amsterdam, Netherlands
- **Pancerz, Krzysztof**, University of Rzeszow, Poland
- **Skowron, Andrzej**, Systems Research Institute, Polish Academy of Sciences and Cardinal Stefan Wyszyński University, Warsaw, Poland
- **Stell, John**, University of Leeds, United Kingdom
- **Suraj, Zbigniew**, Rzeszów University, Poland
- **Szczuka, Marcin**, Institute of Informatics, The University of Warsaw, Poland
- **Yao, Yiyu**, University of Regina, Canada
- **Zhu, William**, University of Science and Technology of China

Three-way Learnability: A Learning Theoretic Perspective on Three-way Decision

Andrea Campagner, Davide Ciucci

Dipartimento di Informatica, Sistemistica e Comunicazione,
 University of Milano–Bicocca, Viale Sarca 336/14, 20126 Milano, Italy

Abstract—In this article we study the theoretical properties of Three-way Decision (TWD) based Machine Learning, from the perspective of Computational Learning Theory, as a first attempt to bridge the gap between Machine Learning theory and Uncertainty Representation theory. Drawing on the mathematical theory of orthopairs, we provide a generalization of the PAC learning framework to the TWD setting, and we use this framework to prove a generalization of the Fundamental Theorem of Statistical Learning. We then show, by means of our main result, a connection between TWD and selective prediction.

I. INTRODUCTION

IN the recent years, there has been an increasing interest toward exploring the connections between learning theory and different uncertainty representation theories: This trend includes both the generalization of standard learning-theoretic tools and techniques to settings that involve representation formalisms that are more general than probability theory [1], [2], as well as the theoretical study of algorithms inspired by uncertainty representation [3], [4].

Among other uncertainty representation theories, Three-way decision (TWD) is an emerging computational paradigm, first proposed by Yao in Rough Set Theory [5], based on the simple idea of *thinking in three “dimensions”* (rather than in binary terms) when representing and managing computational objects [6]: in the Machine Learning (ML) [7] setting, this notion is usually declined in terms of allowing ML models to *abstain*. This approach attracted a large interest, also justified by promising empirical results in different ML tasks such as active learning [8], [9], cost-sensitive classification [10], clustering [11], [12], [9]. Despite these promising empirical results, the theoretical foundations of TWD-based ML received so far little attention [13], [14]. Indeed, even though, in the recent years, there has been an increasing interest toward generalizing computation learning theory (CLT) to cautious inference methods such as *selective prediction* [15] or the KWIK (*Knows what it Knows*) framework [16], such results cannot be easily applied to the TWD setting: While in the TWD setting abstention is a property of single classifiers; in the latter two frameworks abstention is usually achieved by consensus voting.

In this article, we study the generalization of a standard CLT mathematical framework, the so-called *Probably Approximately Correct* (PAC) learning framework, to the TWD setting: In particular, we will provide a generalization of the *Fundamental Theorem of Learning* to the TWD setting, and we

show that our result generalizes previous results in the selective prediction setting. More in detail, the rest of this article is structured as follows: In Section II we provide the necessary mathematical background on TWD (in Section II-A) and CLT (in Section II-B); in Section III we describe the generalization of the PAC learning framework to the TWD setting and we prove our main result; finally, in Section IV, we summarize our contribution and describe possible research directions.

II. BACKGROUND

A. Three-way Decision and Orthopairs

In this work we will refer to the formalization of TWD-based ML models (in the following, TW Classifiers) as *orthopairs*:

Definition 1. An orthopair [17] over the universe X (which represents the instance space) is a pair of sets $O = (P, N)$ such that $P, N \subseteq X$ and $P \cap N = \emptyset$, with P and N standing, respectively, for positive and negative. The boundary is defined as $Bnd = (P \cup N)^c$.

An orthopair represents an uncertain concept: Specifically, the status of the elements in the boundary is uncertain (i.e., it is not known whether they belong to the concept). Thus, a given orthopair stands as an approximation for a collection of consistent concepts:

Definition 2. We say that an orthopair $O = (P, N)$ is consistent with a concept $C \subset X$ if $x \in P \implies x \in C$ and $x \in N \implies x \notin C$.

Finally, we remark that it is possible to define different orderings between orthopairs: In particular, O_2 is *less informative* than O_1 , denoted $O_2 \leq_I O_1$ if $P_2 \subseteq P_1$ and $N_2 \subseteq N_1$.

B. Computational Learning Theory

Computational Learning Theory [18] (CLT) refers to the branch of Machine Learning and Theoretical Computer Science focusing on the theoretical study of learning algorithms. Various mathematical formalisms have been proposed toward this goal, in this article we will refer to the PAC (probably approximately correct) learning framework, first proposed in [19]. Formally, let X be the instance space and Y be the target space, in this article we will focus on the *binary classification* setting, that is $Y = \{0, 1\}$. We assume that the observable data is generated i.i.d. according to an unknown probability distribution \mathcal{D} over $X \times Y$. Let \mathcal{H} be a hypothesis

class, that is a collection of functions $h : X \mapsto Y$, we define the *true risk* of h w.r.t. \mathcal{D} as:

$$\mathcal{L}_{\mathcal{D}}(h) = \mathbb{E}_{\mathcal{D}} [l(h(x), y)] = \int_{X \times Y} l(h(x), y) d\mathcal{D}(x, y) \quad (1)$$

where $l : Y^2 \mapsto \mathbb{R}^+$ is a loss function. Since \mathcal{D} is unknown, the true risk cannot be computed: It is usually approximated through the so-called *empirical risk* based on a sample, called *training set*, $S = (\langle x_1, y_1 \rangle, \dots, \langle x_m, y_m \rangle)$:

$$L_S(h) = \frac{1}{m} \sum_{i=1}^m l(h(x_i), y_i) \quad (2)$$

Given a training set S , we denote by S_X the tuple $S_X = (x_1, \dots, x_m)$, and by S_Y the tuple $S_Y = (y_1, \dots, y_m)$. The *Empirical Risk Minimization* w.r.t. the hypothesis class \mathcal{H} is the family of algorithms $ERM_{\mathcal{H}, m} : (X \times Y)^m \mapsto \mathcal{H}$ s.t. $ERM_{\mathcal{H}, m}(S) \in \operatorname{argmin}_{h \in \mathcal{H}} L_S(h)$, where $S = (\langle x_1, y_1 \rangle, \dots, \langle x_m, y_m \rangle)$ is the training set.

The *Fundamental Theorem of Learning* [20] establishes a relation between the true risk and empirical risk for the *ERM* algorithm w.r.t. a hypothesis class \mathcal{H} which depends only on the so-called VC dimension, a combinatorial dimension of the complexity of \mathcal{H} .

Theorem 1. *Let \mathcal{H} be a hypothesis class with VC dimension d . For each $\epsilon, \delta \in (0, 1)$ and distribution \mathcal{D} , then if $ERM_{\mathcal{H}}$ is given a dataset S of size $m \geq n_0$, with*

$$n_0 = O\left(\frac{d + \ln(\frac{1}{\delta})}{\epsilon^2}\right) \quad (3)$$

with probability greater than $1 - \delta$, it holds that $|L_{\mathcal{D}}(ERM_{\mathcal{H}}(S)) - L_S(ERM_{\mathcal{H}}(S))| \leq \epsilon$. If, further, the realizability¹ assumption holds, then, if S is a dataset of size $m \geq n_1$, with

$$n_1 = O\left(\frac{d + \ln(\frac{1}{\delta})}{\epsilon}\right) \quad (4)$$

with probability greater than $1 - \delta$, it holds that $L_{\mathcal{D}}(ERM_{\mathcal{H}}(S)) \leq \epsilon$.

Few works have studied the generalization of CLT results to hypothesis that can be described as orthopairs (that is, classifiers that can abstain on selected instances), mainly under the framework of *selective prediction* [21]: In this setting, the goal is to design learning algorithms $\mathcal{A}_{\mathcal{H}, m} : (X \times Y)^m \mapsto \mathcal{O}_{\mathcal{H}}$, where $\mathcal{O}_{\mathcal{H}} \subseteq TW(\mathcal{H})$ (see Eq. (15)), s.t. $L_{\mathcal{D}}(A(S)) = 0$ but $A(S)$ is allowed to abstain on certain instances. This abstention is usually achieved either by the combination of a standard hypothesis $h : X \rightarrow Y$ with a rejection function $r : X \rightarrow \{\perp, \top\}$, or, equivalently, by consensus voting based on a version space $V \subseteq \mathcal{H}$ [21]. As we show in the following sections (specifically, in Section III-A) the setting we consider is a proper generalization of selective prediction. More recently, the application of orthopairs in CLT has been studied in the setting of adversarial machine learning [22], as well as to characterize the generalization

capacity of hypothesis classes under generative assumptions [23]. We note, however, that even though the above mentioned work and the framework we study in this article rely on the representation formalism of orthopairs, the aims of these three frameworks are essentially orthogonal, also in terms of the mathematical techniques adopted: Indeed, while the three-way learning framework we study relies on a generalization of the ERM paradigm, the frameworks studied in [23], [22] rely on a transductive learning approach.

III. THREE-WAY LEARNING

In this Section, we provide a first study of a generalization of standard Computational Learning Theory to the setting of TW Classifiers. As hinted in Section II-A, we will represent a TW Classifier as an orthopair O ; then, a hypothesis space of TW Classifier will be represented as a collection \mathcal{O} of orthopairs over X . In the TWD literature, the risk of a TW Classifier is usually evaluated by means of a cost-sensitive gener-

alization of the 0-1 loss: $l_{TW}(O(x), y) = \begin{cases} 1 & O(x) \perp y \\ \lambda_a & x \in Bnd_O \\ 0 & \text{otherwise} \end{cases}$,

where $\lambda_a \in [0, 0.5]$ is the cost of abstention, and $O(x) \perp y$ is the error case, that is $(x \in P_O \wedge y = 0) \vee (x \in N_O \wedge y = 1)$. Compared to the standard definition of risk adopted in the TWD literature we assume that the cost of error is always 1. Based on the loss function l_{TW} we can define both the true risk $\mathcal{L}_{\mathcal{D}}^{TW}$ and the empirical risk L_S^{TW} . Evidently, the risk of O can be decomposed as the sum of two functions:

$$\begin{aligned} \mathcal{L}_{\mathcal{D}}^{TW}(O) &= \mathbb{E}_{\mathcal{D}} [l_{TW}(O(x), y)] \\ &= \mathbb{E}_{\mathcal{D}} [\mathbb{1}_{O(x) \perp y}] + \lambda_a \mathbb{E}_{\mathcal{D}} [\mathbb{1}_{x \in Bnd_O}] \\ &= Pr_{x \sim \mathcal{D}}(O(x) \perp y) \\ &\quad + \lambda_a \cdot Pr_{x \sim \mathcal{D}}(x \in Bnd_O) \end{aligned} \quad (5)$$

The same decomposition can be similarly applied for the empirical risk. Let $\mathcal{E}_{\mathcal{D}}(O) = Pr_{x \sim \mathcal{D}}(O(x) \perp y)$ and $\mathcal{A}_{\mathcal{D}}(O) = \lambda_a \cdot Pr_{x \sim \mathcal{D}}(x \in Bnd_O)$. We denote with $\mathcal{O}^{OPT} = \{O \in \mathcal{O} : \mathcal{E}_{\mathcal{D}}(O) = \min_{O' \in \mathcal{O}} \mathcal{E}_{\mathcal{D}}(O')\}$. We say that \mathcal{D} is *weakly realizable* w.r.t. \mathcal{O} if $\forall O^* \in \mathcal{O}^{OPT}$ it holds that $\mathcal{E}_{\mathcal{D}}(O^*) = 0$. If, furthermore, $\exists O^* \in \mathcal{O}^{OPT}$ s.t. $\mathcal{A}_{\mathcal{D}}(O^*) = 0$, then we say that \mathcal{D} is *strongly realizable*. Through this article, we will assume only weak realizability. Compared to the realizability assumption, weak realizability assumption is indeed much weaker. As an example if the vacuous TW classifier $O_{\perp} = (\emptyset, \emptyset) \in \mathcal{O}$, then every distribution \mathcal{D} is trivially weakly realizable w.r.t. \mathcal{O} , while it is clearly not strongly realizable.

Let $\epsilon \in (0, 1)$, $\alpha \in (0, \lambda_a)$, then $O \in \mathcal{O}$ makes an (ϵ, α) -failure if one of the following holds:

$$\mathcal{E}_{\mathcal{D}}(O) > \epsilon, \quad \mathcal{A}_{\mathcal{D}}(O) > \min_{O \in \mathcal{O}^{OPT}} \mathcal{A}_{\mathcal{D}}(O) + \alpha \quad (6)$$

Thus, O (ϵ, α) -fails if either its error is greater than ϵ , or if its abstention rate is greater, by a margin of at least α , than the lowest abstention rate among those TW Classifiers that make no error. We thus define the notion of *Three-way learnability*:

Definition 3. *\mathcal{O} is Three-way learnable if exists an algorithm $C_m : (X \times Y)^m \mapsto \mathcal{O}$ and $m_{\mathcal{O}} : (0, 1)^2 \times (0, \lambda_a) \mapsto \mathbb{N}$ such*

¹Here realizability means that $\exists h \in \mathcal{H}$ s.t. $L_{\mathcal{D}}(h) = 0$.

that, for each distribution \mathcal{D} , $\epsilon \in (0, 1)$, $\delta \in (0, 1)$, $\alpha \in (0, \lambda_a)$ $\forall m \geq m_{\mathcal{O}}(\epsilon, \delta, \alpha)$, and given $S \sim \mathcal{D}^m$, C returns $O \in \mathcal{O}$, s.t. O (ϵ, α) -fails with probability lower than δ

We then want to provide a characterization for TW learnability, similar to Theorem 1. For this purpose, we first define a generalization of the ERM algorithm to the TWD setting, that we call Three-way Risk Minimization (TW-RM):

Definition 4. Let $S \in (X \times Y)^m$. Then,

$$\begin{aligned} TWRM(S) &= \operatorname{argmax}_{O \in \mathcal{O}} \mathcal{A}_{X \setminus S_X}(O) \text{ s.t.} \\ \mathcal{E}_S(O) &= \min_{O' \in \mathcal{O}} \mathcal{E}_S(O') \\ \mathcal{A}_S(O) &= \min_{O' \in \mathcal{O}^{OPT}} \mathcal{A}_S(O') \end{aligned} \quad (7)$$

Thus, the TWRM algorithm selects, among those TW classifiers with minimal empirical risk, the TW classifier with maximal abstention rate on the non-observed instances (that is, the instances in $X \setminus S_X$). This has the goal of minimizing errors on non-observed instances, and is analogous to the *maximum margin* principle, and the *disagreement coefficient* in version space learning, active learning and selective prediction [15].

In order to characterize TW learnability, given hypothesis class \mathcal{O} (i.e. a collection of orthopairs), we define two derived hypothesis classes. Given any orthopair $O \in \mathcal{O}$ we can define a classifier $h_O : X \mapsto \{0, 1\}$, as: $h_O(x) = \begin{cases} 1 & x \in \text{Bnd}_O \\ 0 & \text{otherwise} \end{cases}$.

We denote the collection of such binary classifiers as $\mathcal{H}_{\mathcal{O}} = \{h_O : O \in \mathcal{O}\}$. Thus, given \mathcal{O} , the derived hypothesis class $\mathcal{H}_{\mathcal{O}}$ describes the abstention capacity of \mathcal{O} : In the classical setting $\mathcal{H}_{\mathcal{O}} = \{h_0\}$, where $\forall x \in X$, $h_0(x) = 0$, as no classifier in \mathcal{O} is able to abstain: For all $O \in \mathcal{O}$, $\text{Bnd}_O = \emptyset$.

In regard to the second derived hypothesis class, we observe that the order \leq_I defined in Section II-A defines a meet semi-lattice [17] on \mathcal{O} with minimal element $O_{\perp} = (\emptyset, \emptyset)$. Then, we denote with $\mathcal{O}^{\top} = \{O \in \mathcal{O} : \nexists O' \in \mathcal{O} \text{ s.t. } O \leq_I O'\}$, i.e. \mathcal{O}^{\top} is the anti-chain of maximally informative elements of \mathcal{O} .

We now prove a generalization of Theorem 1 to the TWD setting, through which we show that the TW learnability of a hypothesis class \mathcal{O} , using the TWRM algorithm, can be characterized in terms of the derived hypothesis classes $\mathcal{H}_{\mathcal{O}}$ and \mathcal{O}^{\top} . In order to do so, we consider the VC dimension of the two derived hypothesis classes $\mathcal{H}_{\mathcal{O}}$ and \mathcal{O}^{\top} as follows:

$$AVC(\mathcal{O}) = VC(\mathcal{H}_{\mathcal{O}}) \quad (8)$$

$$EVC(\mathcal{O}) = \sup\{|S| : S \subseteq X \wedge \forall C \subseteq S \exists O \in \mathcal{O} \quad (9)$$

$$\text{s.t. } C = (P_O \cap S) \wedge (\text{Bnd}_O \cap S) = \emptyset\}$$

Then, the following result holds:

Theorem 2. Let \mathcal{O} be s.t. $AVC(\mathcal{O}) = d_a$ and $EVC(\mathcal{O}) = d_e$. Then, for any distribution \mathcal{D} weakly realizable w.r.t \mathcal{O} , $\epsilon, \delta \in (0, 1)$, $\alpha \in (0, \lambda_a)$, if $TWRM_{\mathcal{O}}$ is given a dataset of size m larger than :

$$O \left(\max \left\{ \frac{1}{\epsilon} \left(d_e + \ln \frac{1}{\delta} \right), \left(\frac{\lambda_a}{\alpha} \right)^2 \left(d_a + \ln \frac{1}{\delta} \right) \right\} \right) \quad (10)$$

then, $TWRM_{\mathcal{O}}(S)$ (ϵ, α) -fails with probability lower than δ .

Proof. We want to guarantee that the following bound holds:

$$\begin{aligned} Pr_{fail} &= P(S : \exists O \in \mathcal{O} \wedge \\ &\quad |\mathcal{E}_D(O) - \mathcal{E}_S(O)| > \epsilon \vee \\ &\quad |\mathcal{A}_D(O) - \mathcal{A}_S(O)| > \alpha) < \delta \end{aligned} \quad (11)$$

Then, the results would follow by uniform convergence. By the union bound, it holds that:

$$\begin{aligned} Pr_{fail} &\leq Pr(S : \exists O \in \mathcal{O}, |\mathcal{E}_D(O) - \mathcal{E}_S(O)| > \epsilon) \\ &\quad + Pr(S : \exists O \in \mathcal{O}, |\mathcal{A}_D(O) - \mathcal{A}_S(O)| > \alpha), \end{aligned} \quad (12)$$

thus, it is sufficient to jointly upper bound the two summands by $\frac{\delta}{2}$. As regards the error rate (i.e \mathcal{E}) bound, we note that:

$$\begin{aligned} Pr(S : \exists O \in \mathcal{O}, |\mathcal{E}_D(O) - \mathcal{E}_S(O)| > \epsilon) \\ Pr(S : \exists O \in \mathcal{O}^{\top}, \mathcal{E}_D(O) > \epsilon) \end{aligned} \quad (13)$$

Since \mathcal{O}^{\top} is a binary hypothesis class, then, by Theorem 1, the above bound holds with probability greater than $1 - \delta$ as long as $|S| \geq \frac{1}{\epsilon} (d_e + \ln \frac{1}{\delta})$. Furthermore, by uniform convergence this holds, in particular, for $TWRM_{\mathcal{O}}(S)$.

For the abstention part, the same line of reasoning can be applied, however, as we only assume weak realizability, only the result in Theorem 1 that applies to agnostic learning can be used. Then, as long as $|S| \geq \left(\frac{\lambda_a}{\alpha}\right)^2 (d_a + \ln \frac{1}{\delta})$ it holds that $|\mathcal{A}_D(O) - \mathcal{A}_S(O)| < \alpha$ with probability greater than $1 - \delta$. This holds, in particular for $TWRM_{\mathcal{O}}(S)$, and thus the theorem follows by uniform convergence and Eq. (12). \square

As a simple corollary, in the strong realizable setting, it can be easily verified that:

Corollary 1. Let \mathcal{O} be s.t. $AVC(\mathcal{O}) = d_a$ and $EVC(\mathcal{O}) = d_e$. Then, for any distribution \mathcal{D} strongly realizable w.r.t \mathcal{O} , $\epsilon, \delta \in (0, 1)$, $\alpha \in (0, \lambda_a)$, if $TWRM_{\mathcal{O}}$ is given a dataset of size m larger than :

$$O \left(\max \left\{ \frac{1}{\epsilon} \left(d_e + \ln \frac{1}{\delta} \right), \frac{\lambda_a}{\alpha} \left(d_a + \ln \frac{1}{\delta} \right) \right\} \right) \quad (14)$$

then, $TWRM_{\mathcal{O}}(S)$ (ϵ, α) -fails with probability lower than δ .

Note that, if $|\mathcal{O}| < \infty$, then it can be easily shown that $AVC(\mathcal{O}) \leq \log_2(\mathcal{H}_{\mathcal{O}})$. Furthermore, it also holds that $EVC(\mathcal{O}) \leq \log_2(\mathcal{O}^{\top})$, as if O satisfies Eq. (8), then it obviously holds that $\text{Bnd}_O = \emptyset$ and hence $O \in \mathcal{O}^{\top}$.

A. Three-way Learning and Selective Prediction

Finally, we show that the proposed mathematical framework and the obtained results can be used to establish a connection between TWD and *selective prediction*. This result relies on the connection between version space theory and orthopairs [17], and allows us to derive a generalization bound, originally proven by El-Yaniv et al. [21], for selective prediction: This shows that the latter setting can be understood as a special case of TWD. Let \mathcal{H} be a hypothesis class of binary classifiers, we call the Three-way Closure of \mathcal{H} , denoted as $TW(\mathcal{H})$, the hypothesis space obtained as:

$$TW(\mathcal{H}) = \bigcup \{O_H : H \subseteq \mathcal{H}\} \quad (15)$$

where, for each $H \subseteq \mathcal{H}$, $O_H = (\{x : \forall h \in H.h(x) = 1\}, \{x : \forall h \in H.h(x) = 0\})$. Basically, we associate with each possible version space H in \mathcal{H} a corresponding orthopair O_H which abstains on every instance for which the hypotheses in H disagree [17]. Then we can prove the following result:

Corollary 2. *Let $|\mathcal{H}| < \infty$, let $\mathcal{O} = TW(\mathcal{H})$ the Three-way Closure of \mathcal{H} , and let $\lambda_a = 1$. Then, for any distribution \mathcal{D} strongly realizable w.r.t \mathcal{O} , and for any $\delta \in (0, 1)$, if $TWRM_{\mathcal{O}}$ is given a dataset of size m , then:*

- 1) *With probability 1 it holds that $\mathcal{E}_{\mathcal{D}}(TWRM_{\mathcal{O}}(S)) = 0$;*
- 2) *With probability greater than $1 - \delta$ it holds that:*

$$A_{\mathcal{D}}(TWRM_{\mathcal{O}}(S)) \leq O\left(\frac{1}{m} \ln\left(\frac{|\mathcal{H}_{\mathcal{O}}|}{\delta}\right)\right) \quad (16)$$

$$= O\left(\frac{1}{m} \left(|\mathcal{H}| + \ln\frac{1}{\delta}\right)\right) \quad (17)$$

Proof. The first equality easily follows from strong realizability and by noting that, by definition of $TW(\mathcal{H})$, $x \notin Bnd_{TWRM_{\mathcal{O}}(S)}$ iff $(x \in S_X \vee \exists v \in \{0, 1\}.\forall h \in \{h' \in \mathcal{H} : \mathcal{E}_S(h) = 0\}, h(x) = v)$. In regard to the second statement, the first inequality follows by standard algebraic manipulations. The equality, on the other hand, follows by noting that $|\mathcal{H}_{\mathcal{O}}| = 2^{|\mathcal{H}|}$ (as $TW(\mathcal{H})$ contains a TW classifier for each possible subset of hypotheses in \mathcal{H}). \square

IV. CONCLUSION

In this article, we aimed at providing an initial study on the generalization of CLT results to the TWD setting. To this purpose, we first proposed an extension of the standard PAC learning framework to the TWD setting, that we called Three-way Learning and showed that our results generalize the previously known results in the selective prediction literature. As our results represent only a first direction in the theoretical study of TWD as applied to Machine Learning, we believe that the following questions would be of particular interest:

- Our analysis in Theorem 2 relies on a generalization of the VC dimension to the TWD setting. Tighter bounds can usually be obtained by relying on concepts such as Rademacher complexities or covering numbers [18]. How can these be generalized to TWD?
- In Corollary 2 we proved that, in the realizable case, selective prediction can be understood as a special case of TWD learning. Does this analysis also applies to the agnostic (i.e. non-realizable) setting [15]?
- PAC-Bayes bounds [24] study generalization bounds that apply when a probability distribution is defined over the hypothesis space. How can the PAC-Bayes framework be generalized to TWD? Interestingly, a very similar open problem has recently been posed also in Belief Function Theory (BFT) [25]. Due to the connection with random sets, a belief function can be seen as a probability distribution over orthopairs [26]: Then, the generalization of the PAC-Bayes framework to TWD would also enable studying the relationships between TWD and BFT.

REFERENCES

- [1] E. Hüllermeier, "Learning from imprecise and fuzzy observations: Data disambiguation through generalized loss minimization," *International Journal of Approximate Reasoning*, vol. 55, no. 7, pp. 1519–1534, 2014.
- [2] G. Ma, F. Liu, G. Zhang, and J. Lu, "Learning from imprecise observations: An estimation error bound based on fuzzy random variables," in *2021 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. IEEE, 2021, pp. 1–8.
- [3] S. Abbaszadeh and E. Hüllermeier, "Machine learning with the sugeno integral: The case of binary classification," *IEEE Transactions on Fuzzy Systems*, 2020.
- [4] E. Hüllermeier and A. F. Tehrani, "On the vc-dimension of the choquet integral," in *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*. Springer, 2012, pp. 42–50.
- [5] Y. Yao, "Three-way decision: an interpretation of rules in rough set theory," in *International Conference on Rough Sets and Knowledge Technology*. Springer, 2009, pp. 642–649.
- [6] M. Ma, "Advances in three-way decisions and granular computing," *Knowl.-Based Syst.*, vol. 91, pp. 1–3, 2016.
- [7] M. Hu, "Three-way bayesian confirmation in classifications," *Cognitive Computation*, pp. 1–20, 2021.
- [8] F. Min, S.-M. Zhang, D. Ciucci, and M. Wang, "Three-way active learning through clustering selection," *International Journal of Machine Learning and Cybernetics*, pp. 1–14, 2020.
- [9] H. Yu, X. Wang, G. Wang, and X. Zeng, "An active three-way clustering method via low-rank matrices for multi-view data," *Information Sciences*, vol. 507, pp. 823–839, 2020.
- [10] H. Li, L. Zhang, X. Zhou, and B. Huang, "Cost-sensitive sequential three-way decision modeling using a deep neural network," *International Journal of Approximate Reasoning*, vol. 85, pp. 68–78, 2017.
- [11] M. K. Afridi, N. Azam, and J. Yao, "Variance based three-way clustering approaches for handling overlapping clustering," *International Journal of Approximate Reasoning*, vol. 118, pp. 47–63, 2020.
- [12] P. Wang and Y. Yao, "Ce3: A three-way clustering method based on mathematical morphology," *Knowledge-based systems*, vol. 155, pp. 54–65, 2018.
- [13] A. Campagner, F. Cabitza, P. Berjano, and D. Ciucci, "Three-way decision and conformal prediction: Isomorphisms, differences and theoretical properties of cautious learning approaches," *Information Sciences*, vol. 579, pp. 347–367, 2021.
- [14] A. Campagner and D. Ciucci, "A formal learning theory for three-way clustering," in *International Conference on Scalable Uncertainty Management*. Springer, 2020, pp. 128–140.
- [15] R. Gelbhart and R. El-Yaniv, "The relationship between agnostic selective classification, active learning and the disagreement coefficient," *J. Mach. Learn. Res.*, vol. 20, no. 33, pp. 1–38, 2019.
- [16] L. Li, M. L. Littman, T. J. Walsh, and A. L. Strehl, "Knows what it knows: a framework for self-aware learning," *Machine learning*, vol. 82, no. 3, pp. 399–443, 2011.
- [17] D. Ciucci, "Orthopairs and granular computing," *Granular Computing*, vol. 1, no. 3, pp. 159–170, 2016.
- [18] S. Shalev-Shwartz and S. Ben-David, *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.
- [19] L. G. Valiant, "A theory of the learnable," *Communications of the ACM*, vol. 27, no. 11, pp. 1134–1142, 1984.
- [20] V. Vapnik, "On the uniform convergence of relative frequencies of events to their probabilities," in *Doklady Akademii Nauk USSR*, vol. 181, no. 4, 1968, pp. 781–787.
- [21] R. El-Yaniv *et al.*, "On the foundations of noise-free selective classification," *Journal of Machine Learning Research*, vol. 11, no. 5, 2010.
- [22] S. Goldwasser, A. T. Kalai, Y. Kalai, and O. Montasser, "Beyond perturbations: Learning guarantees with arbitrary adversarial test examples," *Advances in Neural Information Processing Systems*, vol. 33, pp. 15 859–15 870, 2020.
- [23] N. Alon, S. Hanneke, R. Holzman, and S. Moran, "A theory of pac learnability of partial concept classes," *arXiv preprint arXiv:2107.08444*, 2021.
- [24] O. Rivasplata, I. Kuzborskij, C. Szepesvári, and J. Shawe-Taylor, "Pac-bayes analysis beyond the usual bounds," *arXiv preprint arXiv:2006.13057*, 2020.
- [25] F. Cuzzolin, *The geometry of uncertainty*. Springer, 2017.
- [26] Y. Yao and P. Lingras, "Interpretations of belief functions in the theory of rough sets," *Information sciences*, vol. 104, no. 1-2, pp. 81–106, 1998.

On Multiplicative, Additive and Qualitative Pairwise Comparisons

Ryszard Janicki

Department of Computing and Software
 McMaster University
 Hamilton, ON, L8S 4K1, Canada
 Email: janicki@mcmaster.ca

Mahmoud Mahmoud

IBM Canada
 Markham, Ontario, L6G 1C7, Canada
 Email: mahmoud.mahmoud@usask.ca

Abstract—A relationship between the classical multiplicative pairwise comparisons that are based on $a_{ij}a_{ji} = 1$, the additive model based on $b_j^i + b_i^j = 1$, and qualitative pairwise comparisons that uses the relations $\approx, \sqsubset, \subset, <$ and \prec , is discussed in detail. A special attention is paid to the concept of consistency and weights calculations. An on-line tool is also discussed.

Index Terms—pairwise comparisons, consistency, weights calculation, qualitative judgements

I. INTRODUCTION

THE *pairwise comparisons* method is based on the observation that it is much easier to rank the importance of *two* objects than it is to rank the importance of *several* objects. This very old idea goes back to Ramon Llull in the end of XIII century. Its modern version is due to 1785 influential paper by Marquis de Condorcet and was later developed by Fechner (1860) and Thurstone (1927) [9], [18]. The modern version is usually associated with Saaty’s AHP (Analytical Hierarchy Process) [17].

Classical pairwise comparisons can be called *multiplicative* [6] as a coefficient a_{ij} is interpreted as an entity E_i is a_{ij} times preferred than an entity E_j . Alternatively we may define *additive* pairwise comparisons where a coefficient b_j^i is interpreted as b_j^i measures the importance of E_i in comparison with E_j assuming that their total importance is 1.0 (or 100%) [3], [6].

When mostly subjective judgment is involved, providing immediately reasonable quantitative relationship between two entities is usually difficult if not almost impossible. We usually start with some qualitative (relational) judgment like ‘ E_i is only slightly better than E_j ’, etc., so we need some good qualitative (relational) model as well [5], [9], [8]. In many cases the use of *combined* pairwise comparisons, that involve simultaneous use of multiplicative, additive and qualitative versions is the best and recommended solution [6], [7], [16].

In this paper we will provide a detailed comparison of multiplicative and additive pairwise comparisons, including the concept of consistency and optimal weights assignment. Since in reality, practically every pairwise comparisons process starts with some qualitative estimations, the qualitative pairwise comparisons [5], [9], [8] and their relationship to multiplicative and additive models is also discussed in detail. We will also provide an easy to use on-line tool, called PiXR,

that makes use of the method presented in the paper, rather easy.

II. MULTIPLICATIVE AND ADDITIVE PAIRWISE COMPARISONS

Let E_1, \dots, E_n be a finite set of objects (entities) to be judged and/or analyzed. The quantitative relationship between entities E_i and E_j is represented by a positive number a_{ij} . We assume $a_{ij} > 0$ and $a_{ij} = \frac{1}{a_{ji}}$, for $i, j = 1, \dots, n$ (which implies $a_{ii} = 1$ for all i). If $a_{ij} > 1$ then E_i is more important (preferred, better, etc.) than E_j and a_{ij} is a measure of this relationship (the bigger a_{ij} , the bigger the difference), if $a_{ij} = 1$ then E_i and E_j are indifferent. We call this model *multiplicative* since a_{ij} is interpreted as E_j is a_{ij} times preferred (more important, etc.) than E_j .

The matrix of such (multiplicative) relative comparison coefficients, $A = [a_{ij}]_{n \times n}$, is called a (multiplicative) *pairwise comparison matrix* [17].

A pairwise comparison matrix $A = [a_{ij}]_{n \times n}$ is *consistent* [17] if and only if

$$a_{ij}a_{jk} = a_{ik}, \quad (1)$$

for $i, j, k = 1, \dots, n$. Saaty’s Theorem [17] states that a pairwise comparison matrix A is consistent if and only if there exist positive numbers w_1, \dots, w_n such that $a_{ij} = w_i/w_j$, $i, j = 1, \dots, n$. The values w_i are unique up to a multiplicative constant. They are often called *weights* and interpreted as a measure of importance. Weights may be scaled to $w_1 + \dots + w_n = 1$ (or 100%) and they obviously create ‘natural’ ranking ($E_i < E_j \iff w_i < w_j$ and $E_i \approx E_j \iff w_i = w_j$). In practice, the values a_{ij} are very seldom consistent so some measurements of inconsistency are needed. Saaty [17] proposed an inconsistency index based on the value of the largest eigenvalue of A . The basic problem is that *it does not give any clue where most inconsistent values of A are located* [2], [9], [10]. On the other hand, *distance-based inconsistency* [10], for a given $A = [a_{ij}]_{n \times n}$, defined as:

$$cm_A = \max_{(i,j,k)} \left(\min \left(\left| 1 - \frac{a_{ij}}{a_{ik}a_{kj}} \right|, \left| 1 - \frac{a_{ik}a_{kj}}{a_{ij}} \right| \right) \right) \quad (2)$$

localizes the most inconsistent triad, so we can reduce inconsistency by some minor changes a_{ij}, a_{ik}, a_{kj} . Recently a fast algorithm for inconsistency reduction has been proposed [11].

To find suitable values of weights from an inconsistent, but with acceptable level of inconsistency, matrix A , one can either calculate the principal eigenvector of the matrix A [17], or use the geometric means of columns (or equivalently, rows) of the matrix A [1] (i.e. for $i = 1, \dots, n$, $w_i = \sqrt[n]{\prod_{j=1}^n a_{ij}}$). For small values of the inconsistency index, both methods produce very similar results [2].

When applying pairwise comparisons to various problems we have noticed that, especially when the entities E_i and E_j were not much different, experts felt often much more comfortable and more confident when they were asked to divide 100 quality points between entities E_i and E_j than to provide multiplicative relationship [6], [7], [16], i.e. ratio a_{ij} . Dividing of 100 between E_i and E_j means that we are replacing the multiplicative relationship $a_{ij}a_{ji} = 1$, with the additive relationship $b_j^i + b_i^j = 1$.

In this approach, we model the mutual relationship between E_i and E_j by two numbers b_j^i and b_i^j , where: b_j^i measures the importance of E_i in comparison with E_j assuming that their total importance is 1.0 (or 100%), and similarly for b_i^j . Formally we assume that, for all $i, j = 1, \dots, n$,

$$b_j^i \geq 0, b_i^j \geq 0 \text{ and } b_j^i + b_i^j = 1 \quad (3)$$

Clearly $b_i^i = 0.5$ (or 50%) for all $i = 1, \dots, n$.

The matrix of such additive relative comparison coefficients, $B = [b_j^i]_{n \times n}$, is called an *additive pairwise comparison matrix*.

When b_j^i is interpreted as the probability that judges would prefer the entity E_i over E_j , the equation (3) is exactly the same as in Bradley-Terry model [3].

To transform additive model into standard multiplicative one, we need a mapping $\phi : \langle 0, 1 \rangle \rightarrow \langle 0, \infty \rangle$ such that $a+b = 1$ implies $\phi(a) \cdot \phi(b) = 1$. When a and b are interpreted as *two parts of one whole* (which equals 1.0, or 100%), then $\frac{a}{b}$ represents *ratio* between a and b , and $\frac{a}{1-a}$ represents ratio between a and its complement.

Hence, the most natural mapping seems to be $\phi(a) = \frac{a}{1-a}$. This mapping has many different applications [13], and in our case leads to the transformation of 'additive' model into 'multiplicative' model [6], [9], [8].

For all $i, j = 1, \dots, n$:

$$a_{ij} = \frac{b_j^i}{1 - b_j^i} = \frac{b_j^i}{b_i^j} \quad (4)$$

From equation (4) we immediately get that for all $i, j = 1, \dots, n$, we have:

$$b_j^i = \frac{a_{ij}}{a_{ij} + 1} \text{ and } a_{ij}a_{ji} = 1 \iff b_j^i + b_i^j = 1 \quad (5)$$

We may now analyze and reduce inconsistency by using the formulas for multiplicative case.

III. QUALITATIVE AND COMBINED PAIRWISE COMPARISONS

Instead of numerical values a_{ij} or b_j^i , the binary relations $\approx, \sqsubset, \subset, <, \prec$ and $\sqsupset, \supset, >, \succ$ over the set of entities Ent =

$\{E_1, \dots, E_n\}$ are used [5], [8]. The relations are interpreted as

- $a \approx b$: a and b are *indifferent*,
- $a \sqsubset b$: *slightly in favor* of b ,
- $a \subset b$: *in favour* of b ,
- $a < b$: b is *strongly better*,
- b is *extremely better*.

The tuple $(Ent, \approx, \sqsubset, \subset, <, \prec)$ is called *qualitative pairwise comparisons systems*. The number of relations has been limited to five because of the known restrictions of human mind when it comes to subjective judgments [4], [15]. The above relations are disjoint and cover all the cases and the relation \approx is symmetric and includes identity.

In this case the consistency is defined by a set of 45 axioms [8] that consider all relational compositions of the above relations. The idea behind all these axioms is very simple and natural:

composition of relations should be relatively continuous and must not change preferences in a drastic way.

Consider the following composition of preferences: $a \approx b \wedge b \sqsubset c$. What relationship between a and c is consistent? Intuitively, $a \approx c$ and $a \sqsubset c$ are for sure consistent, $a \subset c$ is debatable, while $a < c$ and $a \prec c$ are definitively inconsistent. This reasoning leads to the Axiom 2.1 [9]:

2.1 $(a \approx b \wedge b \sqsubset c) \vee (a \sqsubset b \wedge b \approx c) \implies (a \approx c \vee a \sqsubset c \vee a \subset c)$. There are two algorithms that remove inconsistency for qualitative pairwise comparisons [8].

Define the relations $\hat{\succ} = \prec, \hat{\supset} = \prec \cup <, \hat{\subset} = \prec \cup < \cup \subset$ and $\hat{\sqsupset} = \prec \cup < \cup \subset \cup \sqsubset$. If the qualitative pairwise comparison system is consistent [9], [8], then the relations $\hat{\succ}, \hat{\supset}, \hat{\subset}$ and $\hat{\sqsupset}$ are (sharp) partial orders.

In reality, almost always we start with some qualitative relationship, then we try to assign some reasonable number (a_{ij} or b_j^i) to it, and after this we provide some consistency analysis [6], [7], [16]. In some cases we go back to qualitative relationship and eventually analyze the case again from the beginning [6], [7], [16]. Finding proper numbers corresponding to qualitative relations is usually tricky as a trusted methodology still does not exist. Usually numbers 1, 2, ..., 5 [10] or 1, 2, ..., 10 [17] are proposed as initial attempts, without much justification except limits of human mind [4], [15]. More systematic approach involves the concept of qualitative consistency [9], [8]. It was proven that if the b_j^i 's or a_{ij} 's are assigned to relations R_{ij} as shown in Table I, then the qualitative pairwise comparisons are consistent [9]. In fact, having a result in opposite direction would be more desirable but this is still an open problem. Nevertheless using Table I proved useful as it usually results in decent inconsistency [6], [7], [16].

The columns 1 and 3 contain intervals, to start the process one needs to pick one point from appropriate interval, and in general a choice is not obvious. To help users without experience default values are proposed in columns 2, 4 and 5. The column 2 contains just mean values of each interval, the column 4 contains the values of a_{ij} 's derived from appropriate values of column 2 interpreted as b_j^i 's, and the column 5

TABLE I

RELATIONSHIP BETWEEN *arithmetic*, *geometric* AND *relational* SCALES AS PROPOSED BY JANICKI AND ZHAI [9]. WE HAVE HERE $a_{ij} = \frac{b_j^i}{1-b_j^i}$.

arithmetic scale		geometric scale			rel. scale	Definition of importance (E_i vs E_j)
range of b_j^i		range of a_{ij}			relation	
range	default value	range	default value 1	default value 2	R_{ij}	
0.44-0.55	0.5	0.79-1.27	1.0	1.0	\approx	<i>indifferent</i>
0.56-0.65	0.6	1.28-1.94	1.5	1.6	\sqsubset	<i>slightly in favour</i>
0.66-0.75	0.7	1.95-3.17	2.3	2.6	\sqcup	<i>in favour</i>
0.76-0.85	0.8	3.18-6.14	4.0	4.7	\succ	<i>strongly better</i>
0.86-1.00	0.9	6.15-	9.0	7.0	\succ	<i>extremely better</i>

contains mean values of the intervals from column 3. When transforming initial relations R_{ij} , what value should we attach to appropriate a_{ij} , the one from column 4 (default 1) or from column 5 (default 2)? This problem will be discussed in detail the next section, however the difference is not a big one. When a choice is left to the users, usually default value 2 is used [6], [16].

IV. CONSISTENCY FOR ADDITIVE PAIRWISE COMPARISONS

In this section we will analyze an intuitively natural concept of consistency for additive pairwise comparisons.

Let $B = [b_j^i]_{n \times n}$ be an additive pairwise comparison matrix. Consider b_k^i , b_j^k and b_j^i . We have $b_k^i + b_i^k = b_j^k + b_k^j = b_j^i + b_i^j = 1$. How can we decide if b_k^i , b_j^k and b_j^i are consistent or not? The values of b_k^i and b_j^k alone do not seem to have any reasonable relations to the value of b_j^i . The value of b_k^i just indicates the part of one that is assigned to the entity E_i when E_i is compared with E_k , so it can hardly be compared to or associated with b_j^i . However we may try to use ratios $\frac{b_k^i}{b_i^k}$, $\frac{b_j^k}{b_k^j}$ and $\frac{b_j^i}{b_i^j}$ to define a relationship between the entities E_i , E_k and E_j that could lead to a sound concept of consistency. One might say that if the data represented by the matrix $B = [b_j^i]_{n \times n}$ are consistent then if the ratio of importance E_i to E_j is α_{ij} , then for each triple i, j, k we should have $\alpha_{ik}\alpha_{kj} = \alpha_{ij}$, and this could be used as basis for a formal definition of consistency.

We will say that an additive matrix $B = [b_j^i]_{n \times n}$ is *consistent*, if and only if, for all $i, j, k = 1, \dots, n$, we have

$$\frac{b_k^i}{b_i^k} \cdot \frac{b_j^k}{b_k^j} = \frac{b_j^i}{b_i^j} \tag{6}$$

For any given multiplicative matrix $A = [a_{ij}]_{n \times n}$, let $B(A) = [b_j^i]_{n \times n}$ be derived from A by $b_j^i = \frac{a_{ij}}{a_{ij}+1}$, and for any given additive matrix $B = [b_j^i]_{n \times n}$, let $A(B) = [a_{ij}]_{n \times n}$ be derived from B by $a_{ij} = \frac{b_j^i}{1-b_j^i}$. Now we have $\frac{b_k^i}{b_i^k} \cdot \frac{b_j^k}{b_k^j} = \frac{b_j^i}{1-b_j^i} \cdot \frac{b_j^k}{1-b_j^k} = a_{ik}a_{kj} = \frac{b_j^i}{1-b_j^i} = a_{ij}$.

Hence an additive matrix $B = [b_j^i]_{n \times n}$ is consistent if and only if the multiplicative matrix $A(B) = [a_{ij}]_{n \times n}$ is consistent.

This provides an additional justification for the mappings ϕ, ϕ^{-1} that transform A into B and B into A respectively. It

also supports the combined pairwise comparisons process [6], [7], [16] that uses consistency index of $A(B)$ as a consistency index of B without much explanation.

For both a multiplicative matrix $A = [a_{ij}]_{n \times n}$ and an additive matrix $B = [b_j^i]_{n \times n}$ the weights w_1, \dots, w_n are measures of importance of entities E_1, \dots, E_n . Consider the entity E_i . All information about E_i is stored in the sequence a_{i1}, \dots, a_{in} - in case of matrix A , or b_1^i, \dots, b_n^i - in case of matrix B . For the matrix A , the weight corresponding to E_i is defined as an eigenvalue λ_i of A [17], or a geometric mean $g_i = \sqrt[n]{a_{i1} \cdots a_{in}}$ [1]. When A is consistent $a_{ij} = \frac{\lambda_i}{\lambda_j} = \frac{g_i}{g_j}$, which is one of the justifications of both methods [1], [17]. Now consider the sequence b_1^i, \dots, b_n^i . While the value of a_{ij} can be interpreted as ‘absolute’, due to $b_j^i + b_i^j = 1$, the value of b_j^i is not ‘absolute’, it is ‘relative to sum equal one’. The value of $\frac{b_j^i}{b_i^j}$ is ‘absolute’ so it can be used for weight calculation, but $\frac{b_j^i}{b_i^j} = a_{ij}$.

This means the weights generated by B , similarly as for consistency, are the same as these generated but $A(B)$.

This again provides some justification to the combined pairwise comparisons procedure [6], [7], [16].

Additive and multiplicative pairwise comparisons can be seen as orthogonal approaches to the same problem. This argument is briefly illustrated in Table II. When the difference between importance of E_i and E_j is small, $E_i \approx E_j$ looks as well justified, but still one of them seems to be slightly better, using additive pairwise comparisons, i.e. b_j^i , is superior to multiplicative approach. Dividing one hundred into, say, 53 to E_i and 47 to E_j is trustworthy and usually can have some justifications based on merits. On the other hand a statement like ‘ E_i is 1.13 times better than E_j ’, which is equivalent to ‘53 to 47’ distribution of 100 points, can seldom be trusted or have convincing justification (unless as a derivation from b_j^i). Hence, when the initial qualitative judgment is \approx or \sqsubset , but there is a reason to believe that we might be a little bit more precise, then the use of b_j^i to represent qualitative relationship is superior to the use of a_{ij} . The situation seems to be opposite for the relations $>$ and *succ*. It is much easier to conclude that E_i is about 5 times more important (better, etc.) than E_j than to decide that the points distributions should be ‘83 to 17’ (unless as a derivation from a_{ij}). We claim that when the initial qualitative judgment is $>$ or \succ , but there is a reason

TABLE II
SOME RELATIONSHIPS BETWEEN a_{ij} , b_j^i AND R_{ij} ($a_{ij} = \frac{b_j^i}{1-b_j^i}$ AND $b_j^i = \frac{a_{ij}}{a_{ij}+1}$).

b_j^i	a_{ij}	R_{ij}	b_j^i	a_{ij}	R_{ij}	a_{ij}	b_j^i	R_{ij}	a_{ij}	b_j^i	R_{ij}
0.5	1.0	≈	0.51	0.104	≈	2.0	0.67	⊃	3.0	0.75	⊃
0.52	1.08	≈	0.53	1.13	≈	4.0	0.8	>	5.0	0.83	>
0.54	1.17	≈	0.55	1.22	≈	6.0	0.86	>	7.0	0.88	γ
0.56	1.27	⊃	0.57	1.33	⊃	8.0	0.89	γ	9.0	0.9	γ
0.58	1.38	⊃	0.59	1.44	⊃	10.00	0.91	γ			
0.60	1.5	⊃	0.63	1.7	⊃						
0.66	1.94	⊃	0.7	2.33	⊃						

to believe that we might be a little bit more precise, then the use of a_{ij} to represent qualitative relationship is superior to the use of b_j^i . The relationship \supset is a gray area, no approach seems to be superior to the other.

When the default values for multiplicative case are used, we recommend the default values 1 for the \approx and \sqsupset relationship and the default values 2 for $>$ and \succ (gray cells in Table I). The Table III illustrate the proposed combined approach that involves multiplicative, additive and qualitative pairwise comparisons.

V. DESCRIPTION OF PAIRWISE MATRIX INCONSISTENCY REDUCTION (PiXR)

To help with determining weights for attributes and parameters, pairwise inconsistency reduction is needed. The calculations and methods presented in this paper although simple and computable, may consume a lot of time and effort. Moreover, the number of calculations grows quadratically based on the number of parameters.

PiXR is an online tool that was developed to help with this computation and weight determination [14]. The tool consists of 4 main sections that guide the user through creating parameters, measurements, conversion between multiplicative & additive matrices, and inconsistency reduction. The tool also supports importing and exporting data as CSV files for ease of use.

PiXR operates on the following abbreviated process of inconsistency reduction and weight calculation [6]:

- 1) Pairwise matrix is provided to PiXR by Subject Matter Expert (SME)
 - a) Matrix may be provided in as quantitative (additive or multiplicative) or qualitative pairwise matrix
 - b) Consistency measure of the matrix is computed via equation 7
 - c) Pairwise matrix is reduced using formulas 13 & 14 from [11]
 - d) Consistency measure is then calculated again via equation 7, and SMEs can revise values and repeat the process
 - e) Weights are calculated using the geometric mean of the columns of the reduced matrix

$$w_i = \sqrt[n]{\prod_{j=1}^n a_{ij}} \quad (7)$$

PiXR has 4 main sections:

- 1) *Parameters* – used for configuring names and order of parameters when populating matrices in later sections
- 2) *Parameter relations* (Qualitative) – used to configure relations between parameters using qualitative relations outlined in section III and Table I. Default values from Table I will be used to feed the next section. The user may also convert the qualitative matrix to multiplicative or additive.
- 3) *Quantitative Matrix* – an editable version of the matrix shown to the user and changes to it will be reflected in the computed consistency measure *cm* index. A reduction threshold in the range of 0.00-1.00 (inclusive), may be specified when reducing the matrix.
- 4) *Reduced Matrix and Weights* – displays the reduced matrix and the percentages/importance of the parameters. Reduced matrix and weights may be exported this section as CSVs.

PiXR comes with 3 sample applications/matrices from [6] which may be used to demonstrate the capabilities of the tool. The import/export function may be used to revise the matrices and repeat the matrix reduction process. SMEs may be involved in this iterative process to generate new weights and rankings.

PiXR is written in Typescript and Javascript XML (JSX). Using this tech stack enables the tool to run in the browser, thus saving users the time and effort needed to download and install binaries. One may argue that Typescript and by extension JavaScript to be slow for this kind of computation style since the algorithm is at worst $O(n^3)$. However, in most cases the number n is relatively small. The approach is validated to converge quickly when using real world data [11] and by the theoretical results of [12].

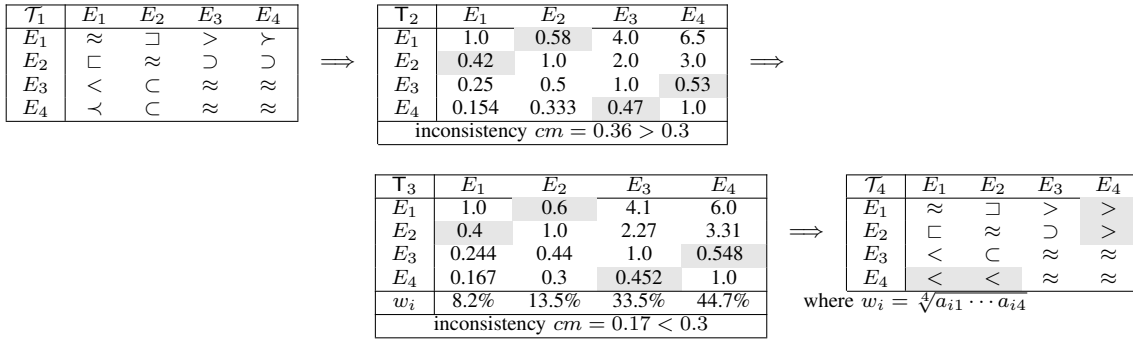
PiXR aims to be *lazy* in most of its evaluations. When computing the triads to measure matrix consistency or computing the localized reduction value, patterns such as generators and streams are used where applicable. This lazy nature allows the user to experiment with the tool and provide as many features as they wish. The computation of the localized inconsistency reduction is also done in a lazy fashion as to not block the main thread and freeze the web-page.

Once the matrix is marked ready for reduction, it goes through the following steps:

- 1) Compute the current consistency measure

TABLE III

A SIMPLE EXAMPLE OF MODIFIED COMBINED PAIRWISE COMPARISONS PROCESS. IN T_2 AND T_3 , GRAY CELLS CONTAIN b_{ij}^j 'S AND WHITE CELLS CONTAIN a_{ij} 'S. IN T_4 , GRAY CELLS INDICATE DIFFERENCES FROM T_1 .



2) While the consistency measure is larger than threshold ϵ , reduce the highest inconsistent triad i, j, k . The threshold ϵ has a default value of 0.3 because it is an acceptable consistency measure with some mathematical justification [10], however it is usually domain dependent.

The above algorithm in pseudo-code is Algorithm 1, where `consistencyMeasure()` is computed using equation 7. Algorithm 2 is the algorithm for reducing the most inconsistent triad. It is based on the results of [11].

Algorithm 1 Inconsistency Index cm

```

cm ← consistencyMeasure(matrix)
while cm ≥ ε do
    reduceTriad(matrix)
    cm ← consistencyMeasure(matrix)
end while
    
```

Algorithm 2 Most Inconsistent Triad

```

ij ← matrix[i][j]
ik ← matrix[i][k]
kj ← matrix[k][j]
a ← ik * kj [Following equation 13 & 14 from [11]]
factor ← 1
if a > ij then
    factor ← -1
end if
b ← factor * (ij + 2 * a)
c ← a - ij
Δc ← min(computeRoots(a, b, c)) [least positive root]
matrix[i][k] ← ik + (factor * ik * Δc)
matrix[k][j] ← kj + (factor * kj * Δc)
matrix[i][j] ← ij - (factor * ij * Δc)
    
```

VI. FINAL COMMENT

A relationships between multiplicative, additive and qualitative pairwise comparisons have been discussed in detail. It was shown that from purely mathematical point of view multiplicative and additive pairwise comparisons are equivalent, but for applications, dependently on the range of data and qualitative relations, one approach can be superior to another. Altogether, a combined approach is recommended. The on-line tool, PiXR, is also provided. The fundamental problem with all methods like the one presented here, is trust. We believe our approach

is more trustworthy than the standard AHP, mainly because we better address the problem of assigning numerical values to subjective qualitative judgements.

REFERENCES

- [1] J. Barzilai, Deriving weights for pairwise comparison matrices, *Journal of the Operational Research Society*, 48, 1226-1232, 1997.
- [2] Bozókí S., Rapcsák T., On Saaty's and Koczkodaj's inconsistencies of pairwise comparison matrices, *Journal of Global Optimization*, 42, 2 (2008), 157-175.
- [3] Bradley R. M., Terry M. E., The rank analysis of incomplete block designs. I. The method of paired comparisons, *Biometrika* 39, 324-345 (1952)
- [4] Cowan N., The magical number four in short-term memory. A reconsideration of mental storage capacity, *Behavioural and Brain Sciences*, 24 (2001) 87-185
- [5] Janicki R., Pairwise Comparisons Based Non-Numerical Ranking, *Fundamenta Informaticae* 94 (2009), 197-217.
- [6] Janicki R., Finding Consistent Weights Assignments with Combined Pairwise Comparisons, *International Journal of Management and Decision Making*, 17, 3 (2018) 322-347.
- [7] Janicki R., Soudkhah M. H., On Classification with Pairwise Comparisons, Support Vector Machines and Feature Domain Overlapping, *The Computer Journal* 58, 3 (2015), 416-431
- [8] Janicki R., Zhai Y., On a Pairwise Comparison Based Consistent Non-Numerical Ranking, *Logic Journal of the IGPL*, 20, 4, 667-676 (2012)
- [9] Janicki R., Zhai Y., Remarks on Pairwise Comparison Numerical and Non-Numerical Rankings, *Lecture Notes in Artificial Intelligence* 6954, Springer 2011, pp. 290-300.
- [10] Koczkodaj W. W., A new definition of consistency of pairwise comparisons, *Mathematical and Computer Modelling*, 18 (1993) 79-84.
- [11] Koczkodaj, W.W., Kosiek, M., Szybowski, J. and Xu D., Fast convergence of distance-based inconsistency in pairwise comparisons', *Fundamenta Informaticae*, 137,3 (2015) 355-367
- [12] Koczkodaj W. W., Szarek S. J., On distance-based inconsistency reduction algorithms for pairwise comparisons, *Logic Journal of the IGPL* 18, 6 (2009) 859-869.
- [13] Kong F., Liu H., Applying fuzzy Analytic Hierarchy Process to evaluate success factors of e-commerce, *International Journal of Information and Systems Sciences*, 1, 3-4 (2005) 406-412.
- [14] Mahmoud M., PiXR: Pairwise Matrix Inconsistency Reduction Tool, <https://pizr-tool.github.io/>
- [15] Miller G. A., The Magical Number Seven, Plus or Minus Two, *The Psychological Review*, 63, 2 (1956) 81-97.
- [16] Mirdad A., Janicki R., Applications of Mixed Pairwise Comparisons, Proc. of ICAI'2015 (International Conference on Artificial Intelligence), Las Vegas, Nevada, USA, July 27-30, 2015, pp. 414-420, CSREA Press
- [17] Saaty T. L., A Scaling Methods for Priorities in Hierarchical Structure, *Journal of Mathematical Psychology*, 15 (1977) 234-281.
- [18] Saaty T. L., *Theory and Applications of the Analytic Network Process*, RWS Publications, Pittsburgh 2005.

Malware Evolution and Detection Based on the Variable Precision Rough Set Model

Manel Jerbi
SMART Lab, CS department
University of Tunis, ISG
 Tunis, Tunisia
 manel.jerbi@gmail.com

Zaineb Chelly Dagdia
Université Paris-Saclay,
 UVSQ, DAVID, France
LARODEC, ISG, Université de
 Tunis, Tunis, Tunisia
 zaineb.chelly-dagdia@uvsq.fr

Slim Bechikh
SMART Lab, CS department
University of Tunis, ISG
 Tunis, Tunisia
 slim.bechikh@fsegn.rnu.tn

Lamjed Ben Said
SMART Lab, CS department
University of Tunis, ISG
 Tunis, Tunisia
 lamjed.bensaid@isg.rnu.tn

Abstract—To offer innovative malware evolution techniques, it is appealing to integrate approaches that handle imperfect data and knowledge. In fact, malware writers tend to target some precise features within the app’s code to camouflage the malicious content. Those features may sometimes present conflictual information about the true nature of the content of the app (malicious/benign). In this paper, we show how the Variable Precision Rough Set (VPRS) model can be combined with optimization techniques, in particular Bilevel-Optimization-Problems (BLOPs), in order to establish a detection model capable of following the crazy race of malware evolution initiated among malware-developers. We propose a new malware detection technique, based on such hybridization, named Variable Precision Rough set Malware Detection (ProRSDet), that offers robust detection rules capable of revealing the new nature of a given app. ProRSDet attains encouraging results when tested against various state-of-the-art malware detection systems using common evaluation metrics.

I. INTRODUCTION

THE amount of malware is increasing exponentially thanks to the use of advanced malware-development tools [1]. Detection models struggle to keep up with these tricky intrusion malicious apps that do not refrain from using the most effective techniques, like the obfuscated malware, to invade the targeted systems. In this course of malware development, one can encounter some data inconsistency specially when dealing with conflictual features that may appear in both benign and malicious apps. In this context, few research has focused on dealing with the inconsistency encountered when extracting relevant features to either produce or detect malware. Authors in [2] proposed to adjust the malware detectors in order to deal with the change that occurs in the data labeling. More precisely, authors tackled the problem that appears when the labels used in the training set are different from the labels used in the testing set and proposed to empirically quantify the epistemic uncertainty of four combined deep-learning based Android malware detectors. Santos et al., in [3], proposed a semi-supervised learning based method to deal with the existing unlabelled apps (unknown nature beforehand) in the training process of a detection process. Also, Nauman et al. [4] looked into a three-way decision-making process

based on acceptance, rejection, or deferment. When there is not enough knowledge, the extra deferment choice option gives the opportunity to postpone a decision. It also seeks to reduce incorrect decisions at the model level by finding a trade-off between decision-making attributes like accuracy, generality, and uncertainty. The authors focused on three-way decisions using two probabilistic rough set models: game-theoretic rough sets (GTRS) and information-theoretic rough sets (ITRS). RoughDroid [5] is a floppy analysis technique proposed by the authors that can detect Android malicious programs straight on the smartphone. It is based on seven feature sets extracted from the XML manifest file of an Android application and three feature sets extracted from the Dex file. Those feature sets pass through the Rough Set algorithm to classify the app either as benign or malicious. In this paper, we propose to specially focus on malware motif production and handling the “false” produced ones that may lead to data inconsistency. We genuinely propose to handle this challenging task and address it by evolving effective malicious motifs, a succession of frequent Application Programming Interface (API) call sequences, and exploit them afterwards in a bilevel-based method in order to produce detection rules capable of detecting them. In this work, we aim to attend the following contributions:

- Generate fraudulent motifs and then exploit them to produce robust detection rules by adopting a bilevel architecture where two Evolutionary Algorithms (EAs), an outer one (Genetic Programming algorithm (GP)) and an inner one (Genetic Algorithm (GA)), are in a mutual competition.
- Inspect the generated fraudulent motifs, which are generated by the inner algorithm within the second layer using the GA, using the Variable Precision Rough Set (VPRS) model before sending them to the outer algorithm within the first layer, i.e., the GP.
- Demonstrate the benefits of the selection made by VPRS reinforced by the bi-level competition between both algorithms since for every detection rule, there exists a whole search space of possible generated malicious motifs that should be effectively sampled to come up with fit and

This work is supported by ANR PIA funding: ANR-20-IDEE-0002.

challenging generated motifs that positively affect the detection quality of the corresponding first layer rule.

- Evaluate the outperformance of our ProRSDet approach compared to several state-of-the-art detection methods in terms of accuracy maximization and false alarms minimization.

The remainder of this paper is structured as follows: Section II emphasizes past work that is most similar to our approach. The fundamentals of BLOP and VPRS used in this work are presented in Section III. Our suggested detection method is described in Section IV. The experimental setup and performance analysis results are presented in Section V. Finally, the conclusion and a description of some future directions are presented in Section VI.

II. RELATED WORK

Different malware detection techniques [6], [7], [8] have been proposed in literature focusing, particularly, on generating new malware. These can be categorized into two heads. A first category is based on using machine learning based approaches and second category based on the use of evolutionary algorithms.

Among the works proposed in the first head, we mention the work of [9] where an Android malware detection system (DroidEvolver) was proposed that can automatically update itself during malware detection using online learning techniques with evolving feature set and pseudo labels. There were some methods which were based on generating adversarial samples. Among these, we mention, the work proposed in [10], where the feasibility of generating adversarial samples specifically through the injection of system API calls was investigated. In [11], a generative adversarial network based algorithm (MalGAN) was proposed to generate adversarial malware examples able to bypass black-box machine learning based detection models. The paper of Moti et al. [12] presented a deep generative adversarial network to generate the signature of unseen malware samples; the generated signature is potentially similar to the malware samples that may be released in the future.

Other works proposed automated signature generation systems, such as the work proposed in [13], where a system for automatic generation of intrusion signatures from honey net packet traces was developed. The work of [14], proposed an automated approach called “content sifting” that generates precise signatures that can then be used to filter or moderate the spread of a worm. In [15], a string signature generation system (Hancock) was designed to create a minimal set of N-byte sequences from a set of malware samples. Another work, the work of [16], used a 5-gram Markov chain model of good software to estimate the probability that a given byte sequence would show up in good software. In the paper of Li et al. [17], a network-based automated signature generation system (Hamsa) for polymorphic worms was proposed. The proposed model allowed to analyze the invariant content of polymorphic worms in order to make analytical attack-resilience granted for the signature generation algorithm. In

[18], Newsome et al. proposed a signature generation system, Polygraph, that produces signatures that match polymorphic worms. Polygraph generates signatures that consist of multiple disjoint content sub-strings and which typically correspond to protocol framing, return addresses, and poorly obfuscated code.

Within the second head, several works [19], [20], [21], [22], [23], focused on applying evolutionary algorithms to generate malware samples. Among the most recent and efficient ones, we mention the work of [24], where an Android Malware Detection System (AMD) was proposed that produces patterns using a GA in order to mimic real malware patterns. This is to keep the dataset used in the conception of the detection system as varied as possible, which allows AMD to be resistant to obfuscated malware. Also, the work of [23], opted for a system using co-evolutionary algorithms where a first population generates detection rules, and a second population generates artificial malware. In this work, both populations are executed in parallel without any hierarchy. In the works of [25], [26], authors adopted a co-evolutionary algorithm as a search engine to ensure better detection rules.

Despite the good reached results of the above mentioned state-of-the-art methods, they still suffer from some limitations. First, they refer to a limited number of malware samples which makes the produced base of malicious malware not varied enough which cannot be of much help for a detection system when facing real attacks. Second, there is no check of the structure of the generated malicious patterns as to be sure enough that they fit among the real samples. And third, the malware generation and detection tasks are achieved separately without interaction which leads to a lack of a “harmony” and hence creates an incompatibility between the tasks.

In in paper, we will introduce our newly developed ProRSDet malware detection technique that overcomes the state-of-the-art shortcomings via the hybridization of both evolutionary algorithms and the Variable Precision Rough Set model.

III. BLOP AND VPRS BASIC CONCEPTS

In this section, we introduce the main concepts and fundamentals of both BiLevel OPTimization and the Variable Precision Rough Set model as two tools, used in a hybrid fashion, to ensure the development of our proposed ProRSDet.

A. BiLevel Optimization

BLOP is a distinctive optimization process where one problem is embedded within another. The inner problem, which is also referred to as the lower-level task, represents a constraint of the outer problem, which is also referred to as the upper-level task, where only an optimal lower-level solution can be a possible solution to the upper-level one. Each level has its own fitness function to optimize where the considered solutions of each level affect the decision-making space of the other one. The technical formalization of a BLOP problem can be found in [27] and it can be presented as follows:

A BLOP contains two classes of variables: (1) the upper-level variables $x \in X \subset R^n$, and (2) the lower-level variables

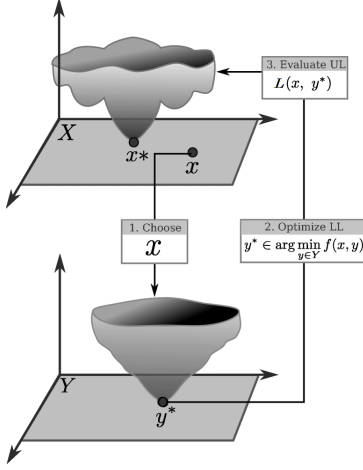


Fig. 1. Representation of a bilevel optimization problem (Inspired by [28])

$y \in Y \subset R^m$. For the follower problem, the optimization task is performed with respect to the variables y while the variables x act as fixed parameters. Thus, each x corresponds to a different follower problem, whose optimal solution is a function on Y and needs to be determined. All variables (x, y) are considered in the leader problem for given values of $y (y^*)$. In what follows, we give the formal definition of BLOP.

Assuming $L : R^n \times R^m \rightarrow R$ to be the leader problem and $f : R^n \times R^m \rightarrow R$ to be the follower one, a BLOP could be defined as follows:

$$\min_{x \in X, y \in X_L} L(x, y) \text{ subject to } \begin{cases} G_k(x, y) \leq 0, k = 1, \dots, K. \\ y \in \operatorname{argmin}\{f(x, y)\} \\ g_j(x, y) \leq 0, j = 1, \dots, J \end{cases} \quad (1)$$

In the given formulation, L represents the first layer objective function, f represents the second layer objective function, x represents the first layer decision vector and y represents the second layer decision vector. G_k and g_j represent the inequality constraint functions at both layers, respectively. The representation of a bilevel optimization problem is illustrated in Figure 1.

B. Variable Precision Rough Set

VPRS [29], an extension of RST, is a mathematical tool that deals with inconsistent information and came mainly to overcome the maybe found strictness within the rough set notions which may be too restricted in the sense that they ignore the degree of an overlap between a set and a concept. Let us consider a universe of objects \mathbf{U} referred to as elementary events and let $s(\mathbf{U})$ be the ∂ -algebra of measurable subsets of \mathbf{U} referred to as random events. It is presumed that new objects e belonging to the universe are generated by a random process (on \mathbf{U}). For each new object e , the event $X \in s(\mathbf{U})$ occurred if the object $e \in X$. In addition, it is presumed the existence of the prior probability function P assigning probabilities $P(X)$ to sets X belonging to $s(\mathbf{U})$. $P(X) > 0$ means that all members of the family of sets $s(\mathbf{U})$ are likely to occur, and, $P(X) < 1$ means that their

occurrence is not certain. These assumptions are justified by the fact that there is no need to construct a predictive model for events about which it is known that they are unlikely to occur or that they do occur with certainty [29]. In the context of defining the structure of rough approximation space, R denotes an equivalence relation on \mathbf{U} with the finite number of equivalence classes (elementary sets) E_1, E_2, \dots, E_n such that $P(E_i) > 0$ for all $1 \leq i \leq n$. The assumption of finite number of equivalence classes does not mean that the universe \mathbf{U} is finite. Each elementary set E can be assigned a measure of overlap with the set X by the conditional probability function defined as $P(X | E) = P(X \cap E) / P(E)$. The values of the conditional probability function are normally estimated from sample data by taking the ratio $P(X | E) = \operatorname{card}(X \cap E) / \operatorname{card}(E)$. The VPRS generalization of the original rough set model is based on the values of the probability function P and two lower and upper limit certainty threshold parameters l and u such that $0 \leq l < P(X) < u \leq 1$. The requirement $l < P(X)$ is an extra constraint on the values of the parameters which was proposed in [29]. The VPRS model is said to be symmetric if $l = 1 - u$. In this study, the symbol β such that $0 < P(X) < \beta \leq 1$ is used instead of the symbol u to denote the model upper threshold parameter. Also, the symbol α will substitute the previously defined l parameter.

IV. PRORSDET: THE VARIABLE PRECISION ROUGH SET MALWARE DETECTION TECHNIQUE

Figure 2 depicts ProRSDet's overall running process, which is divided into two principal layers (levels): (1) *First layer* is built on a GP with the goal of generating a set of effective detection rules (*FDRB*) and (2) *Second layer* relies on a GA to generate harmful (malicious) motifs (*SHM*) (first step) and on a VPRS based component that exclusively preserves the most dependable set of harmful motifs with no structural flaws, referred to as "Relevant" motifs (*FMM*) (second step).

Each of these two layers runs through a series of iterations in order to find the optimal solutions in both levels, which are interdependent. As presented in Figure 2, the evaluation of every upper detection rule solution (among *DRB*) requires running a search algorithm to find the best undetectable harmful motifs (*FMM*) by this rule. The final set of detection rules produced by our ProRSDet (*FDRB*) is a set of detection rules that will perform the malware detection task.

1) *First layer*: The first layer's first step, as shown in Figure 2 and Algorithm 1, is to generate a set of detection rules (Algorithm 1, line 1), which will go through an evaluation procedure (Algorithm 1, lines 2-3). The coverage of the base of samples (input) as well as the coverage of the fraudulent motifs created by the second layer are used to make this evaluation. These two measures are used to be maximized by the population of detection rules solutions (Algorithm 1, lines 4-6). This module produces a collection of final detection rules (*FDRB*) that will be used by the detection job, which is in charge of classifying new apps as malicious or benign. The GP evolutionary operators require a specific formalization to cope

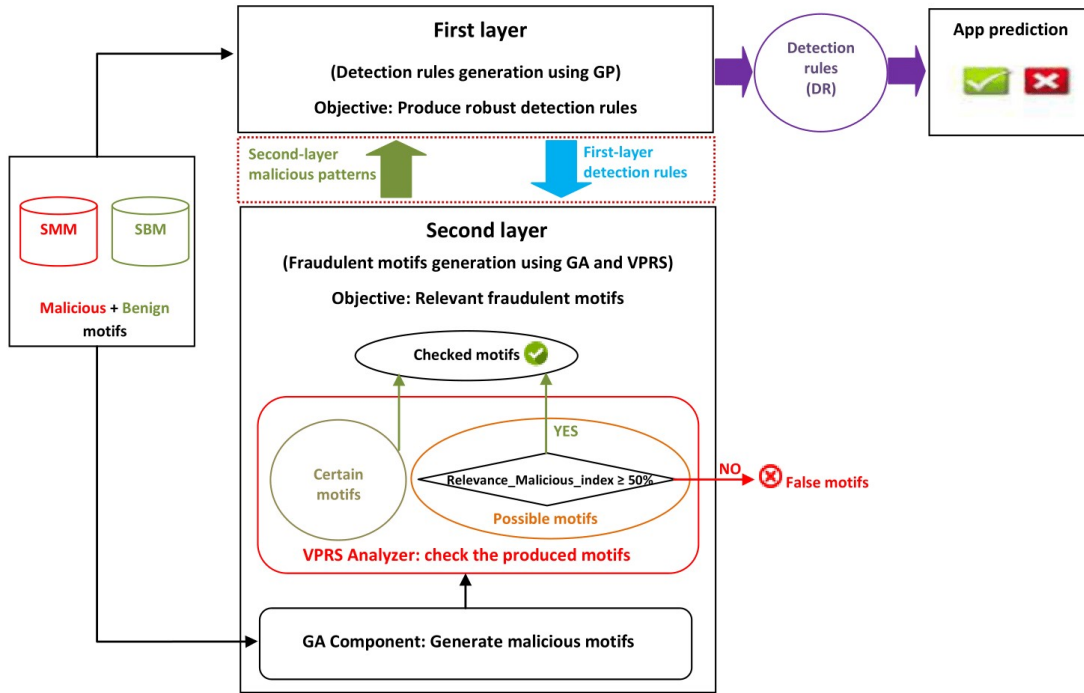


Fig. 2. Illustration of the ProRSDet functioning process.

with the generated solutions (i.e., the detection rules) by the first layer that relies on a GP process. These are the following:

- *Solution representation*: The solution is expressed as a series of terminals that relate to various motifs (API call sequences) and functions (Intersection (*AND*) and Union (*OR*)), respectively.
- *Solution variation*: By selecting one of the functions or terminals at random, the GP *mutation* operator is applied. If a terminal is selected then it is replaced by another terminal; if it is a function then it is replaced by a new function. As for the GP *crossover operator*, two parent individuals are selected, and a sub-node is picked on each selected parent. The *crossover* swaps the nodes and their related sub-nodes from one parent to the other.
- *Solution evaluation*: An individual's encoding is quantified using a mathematical metric called the "fitness function", which measures the quality of a proposed detection rule and fraudulent motifs. For the GP adaptation, we used the fitness function f_{outer} defined in Equation 2 to evaluate detection-rules solutions (*DR*).

$$f_{outer}(DR) = \text{Max}\left(\frac{\text{Precision}(DR) + \text{Recall}(DR)}{2} + \frac{\#damp}{\#amp}\right) \quad (2)$$

where $\#damp$ refers to the number of detected fraudulent motifs and $\#amp$ refers to the number of fraudulent motifs and

$$\text{Precision}(DR) = \frac{\sum_{i=1}^p DR_i}{t}, \text{Recall}(DR) = \frac{\sum_{i=1}^p DR_i}{p} \quad (3)$$

Algorithm 1 Outer Algorithm (First layer)

Input: *SMM*: set of malicious motifs, *SBM*: set of benign motifs, *FMM*: set of "Relevant" fraudulent motifs, *NDR*: number of detection rules, *NFM*: number of "Relevant" fraudulent motifs in *SHM*, *NF*: number of iterations in the first layer, *NS*: number of iterations in the second layer

Output: Final set of detection rules

FDRB

```

1: DRB0 ← Initialization(NDR, SHM, SBM) /*First generation of detection rules*/
2: for each DR0 in DRB0 do /*DR means detection rule*/
   SFM0 ← FMGeneration(DR0, FMM, NFM, NS) /*call second layer*/
   DR0 ← Evaluation(DR0, SHM, SFM0)
4: end for
   k ← 1
6: while k < NF do
   Qt ← Variation(DRBt-1)
8:   for each DRt in Qt do /*Evaluate each rule based on upper fitness function*/
     DRt ← OuterEvaluation(DRt, SHM)
10:    SFMt ← FMGeneration(DRt, SHM, NFM, NS)
     DRt ← EvaluationUpdate(DRt, SFMt)
12:   end for
   Ut ← Qt ∪ DRBt
14:   DRBt+1 ← Selection(NDR, Ut)
   k ← k+1
16: end while
FDRB ← FittestSelection(DRBt)

```

p is the number of detected malicious motifs after executing the solution, i.e., the detection rule, on the base of malicious motifs examples (SMM), t is the total number of malicious motifs within SMM , and DR_i is the i^{th} component of a detection rule DR such that:

$$DR_i = \begin{cases} 1 & \text{if the } i^{th} \text{ detected malicious motif exists in } SMM \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

2) *Second layer*: The generation process of “Relevant” motifs (FMM , Algorithm 2, line 7) is performed as follows:

- *Step 1*: A GA is used to maximize the distance between the generated malicious motifs (SHM) and the reference benign motifs (input, not-generated motifs (SBM) while minimizing the distance between the generated malicious motifs (SHM) and the reference malicious ones (SMM). The GA also increases the amount of malicious motifs generated that are not detected by the first layer, i.e., the detection rules (DRB) (Algorithm 2, lines 1-5). The GA evolutionary operators need a special formalization to deal with the manipulated solutions in order to generate the motifs. The following are the adopted formalizations:

- *Solution representation*: The GA solutions are represented as chromosomes made up of API call sequences. These are identifiable by their identifiers (IDs) and defined by their class (labels), which indicate their nature (malicious or benign), calling depths, and a collection of binary values indicating whether or not an API call appears in the entire API call sequence.
- *Solution variation*: For the GA *crossover operator*, two parent individuals (chromosomes) are chosen, and a gene from each parent is chosen. Crossover involves the transfer of genes from one parent to the other. Only parents with the same nature can be used with the crossover operator (malicious or benign). The *mutation* operation starts by randomly selecting a gene on the chromosome. The selected gene is then replaced with another gene from the same class if it belongs to that class.
- *Solution evaluation*: A fraudulent motif (FM) is evaluated based on the following GA fitness function:

$$f_{inner}(FM) = Max((\#gamp - \#dagmp) + \sum_{i=1}^n f_{Qual}(FM_i)) \quad (5)$$

where $i \in [1, n]$; n indicates the total number of fraudulent motifs, and $\#gamp$ refers to the number of fraudulent motifs and $\#dagmp$ refers to the number of detected fraudulent motifs. The function $f_{Qual}()$ defined in Equation 6 ensures the diversification of the fraudulent motifs.

$$f_{Qual}(FM_i) = \frac{Sim_1 + Sim_2 + Overlap(FM_i)}{3} \quad (6)$$

$$Sim_1 = Sim(MS, FM_i) = \frac{\sum_{MS_j \in MS} Sim(FM_i, MS_j)}{|MS|} \quad (7)$$

where $j \in [1, m]$; m indicates the total number of malicious motifs. The similarity between the generated motif FM_i and the malicious set of motifs (MS). This measure of similarity needs to be maximized.

$$Sim_2 = Sim(BS, FM_i) = \frac{\sum_{BS_k \in BS} Sim(FM_i, BS_k)}{|BS|} \quad (8)$$

The similarity between the generated motif FM_i where $k \in [1, p]$; p indicates and the benign motifs (BS) the total number of benign set of motifs and which has to be the lowest.

$$Overlap(FM_i) = 1 - \frac{\sum_{FM_l, l \neq i} Sim(FM_i, FM_l)}{|FM|} \quad (9)$$

$Overlap()$ is measured as the average value of the individual $Sim(FM_i, FM_l)$ between the generated motif FM_i and all the other generated motifs FM_l in the generated dataset SFM . l refers to the total number of the generated motifs.

We updated the Needleman-Wunsch alignment algorithm formula [30] to our context to determine the similarity $Sim()$ between two motifs. This measure of similarity was employed in the above equations but with different parameters. A detailed description of the similarity function $Sim()$ can be found in [24].

- *Step 2*: The GA evolutionary operators mentioned above may cause the manipulated solutions to be distorted, and hence ambiguous, in different ways and with different degrees. Technically, a set of motifs is declared to be ambiguous when they share the same values of the features (API calls) but do have different label values (malicious/benign). An illustration of this ambiguity is presented in Table I. The manipulated motifs by the lower-level are API call sequences. Each API call sequence is named MFx_i (as shown in Table I) and is composed of different API calls named MLX_j . A conflict (or inconsistency) may exist between objects (fraudulent motifs). It is the case of the objects MFx_7 and MFx_9 because they are indiscernible by condition attributes MLX_1, \dots, MLX_n and have different decision attributes (Nature) (we assume that all attribute values MLX_j are the same). Similarly, another inconsistency exists between objects MFx_3 and MFx_8 .

To handle this ambiguity issue and to guarantee the reliability of the generated malicious motifs, a VPRS component, namely Variable Precision Rough Set Analyzer (VPRS Analyzer in Figure 3) which uses mainly

TABLE I
EXAMPLES OF AMBIGUOUS MOTIFS.

Malicious fraudulent motifs	Condition attributes (API call)				Decision (Nature)
	MLX_1	MLX_2	...	MLX_n	
MFx_1	1	1	...	1	M
MFx_2	0	0	...	0	M
MFx_3	1	0	...	0	M
MFx_4	1	0	...	1	M
MFx_5	1	1	...	0	M
MFx_6	1	0	...	1	M
MFx_7	1	0	...	1	B
MFx_8	1	0	...	0	B
MFx_9	1	0	...	1	M
MFx_{10}	1	1	...	0	B

Algorithm 2 Inner Algorithm (Second layer)

Input: SMM : set of malicious motifs, SBM : set of benign motifs, DRB : set of detection rules, R : number of generations, N : population size

Output: Set of Relevant generated motifs

FMM

```

1:  $SFM_0 \leftarrow \text{Initialization}(SBM, SMM, N, R)$  /*SFM means set of fraudulent motifs*/
2:  $SFM_0 \leftarrow \text{Evaluation}(SFM_0, SBM, SMM, DRB)$  /*Evaluation depends on DRB*/
3:  $k \leftarrow 1$ 
4: while  $k < R$  do
5:    $Q_t \leftarrow \text{Variation}(SFM_{t-1})$ 
6:    $Q_t \leftarrow \text{Evaluation}(Q_t, SBM, SMM, DRB)$ 
7:    $U_t \leftarrow Q_t \cup SFM_t$ 
8:    $SFM_{t+1} \leftarrow \text{Selection}(N, U_t)$ 
9:    $k \leftarrow k+1$ 
10:   $SHM \leftarrow \text{FittestSelection}(SFM_t)$ 
11:   $(RFM, AFM) \leftarrow \text{RelevanceCheck}(SHM)$  /*Set of relevant FM and a set of ambiguous FM*/
12:   $SCFM \leftarrow \text{LowerCertainty}(AFM) \cup RFM$ 
13:   $SPFM \leftarrow \text{UpperCertainty}(AFM)$ 
14:   $(FCFM, FPFM) \leftarrow \text{Pruning}(SCFM, SPFM)$ 
15:   $FMM \leftarrow FCFM \cup FPFM$ 
16: end while

```

the VPRS lower and upper limit certainty thresholds concepts, is plugged to the inner algorithm. Specifically, the VPRS Analyzer checks first the reliability of the generated malicious motifs (SHM). Among this set, the VPRS Analyzer keeps the most relevant motifs (RFM) which do not need any further check, and investigates the remaining ambiguous set (AFM). Among the AFM set, the VPRS Analyzer calculates the lower limit certainty threshold to only keep the certain set of fraudulent motifs $SCFM$ (Algorithm 2, lines 11-12), and the upper limit certainty threshold to keep the approved fraudulent motifs among the *possible* set of generated fraudulent motifs $SPFM$, together with $SCFM$. During the pruning operation (Algorithm 2, line 14), redundant motifs are removed. Finally, the joint sets of $FCFM$ and $FPFM$ form the relevant, and the most relevant artificially generated fraudulent motifs (FMM) (Algorithm 2, line 15).

As *possible* fraudulent motifs cannot be considered relevant enough to be added to the initial set of malicious motifs, they need further evaluation that reflects their quality and measures

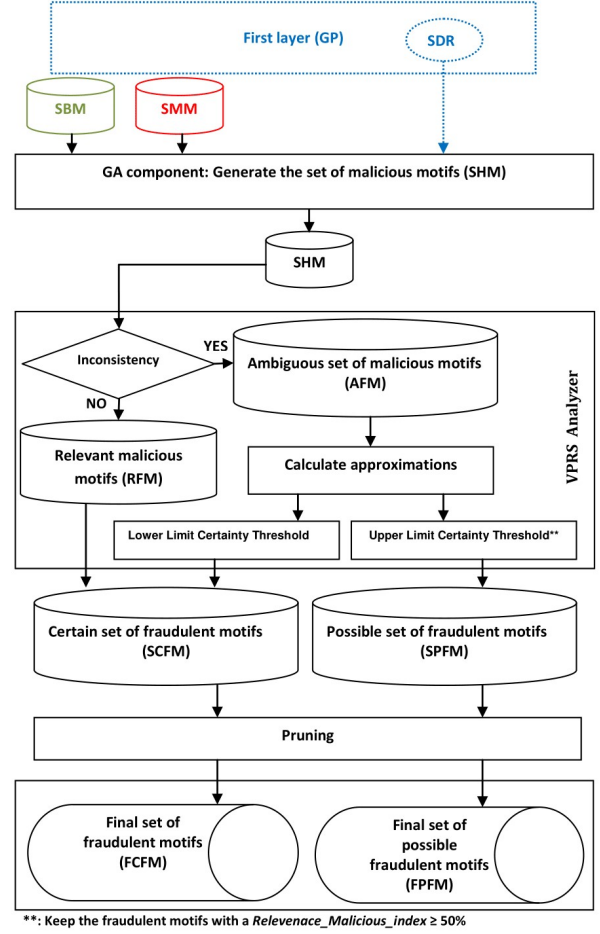


Fig. 3. Second layer functioning process.

their reliability. For every *possible* fraudulent motif, the VPRS-based component estimates its reliability using an index named *Relevance_Malicious_index*, which is defined as the ratio of the number of instances that belong to the *possible* set and having the same structure with a malicious label, and the number of the whole *possible* set of instances (Equation 10).

$$Relevance_Malicious_index = \frac{Instance_Possible_Malicious_Motif}{Instance_Possible_Original_Data} \quad (10)$$

where *Instance_Possible_Malicious_Motif* refers to the number of instances that share the same structure and are labelled malicious and *Instance_Possible_Original_Data* refers to the total number of instances within the whole possible set. This index can, therefore, be viewed as the probability of counting the ambiguous fraudulent motifs set (the *possible* generated motif set) correctly. It shows the extent to which a correct label can be given to a generated motif belonging to the *possible* set of generated motifs. Tacking into account that we are aiming to produce effective malicious fraudulent motifs, we will only keep the generated motifs that have a *Relevance_Malicious_index* > 50%. An illustrative example of this index is given below: Sup-

pose that when generating 123 new fraudulent motifs, 100 among them were labelled as malicious and 23 were labelled as benign. The *Relevance_Malicious_index* of those 123 possible generated motifs is (100/123). This means that the *Relevance_Malicious_index* = 81,30% which is clearly greater than 50% and hence the common shared structure of these generated motifs will be added to the set of malicious motifs sent to the outer algorithm.

3) *Detection task based on detection rules*: Throughout this phase, our model will perform its classification task where a new app, the executable, will be classified either as a malware or as a benign. This is achieved using the set of detection rules (*FDRB*). Formally, the first step aims to extract the motifs of the executable. Each motif will be labeled as benign or as malicious by comparing it to the motifs of the *SMM* and *SBM* databases. Then, the obtained motifs are compared to the antecedent of *FDRB*. The comparison will allow the executable to be either classified as a malware or as a benign app.

V. EXPERIMENTAL SETUP AND RESULTS

A. Experimental Setup

An experimental investigation was done to evaluate ProRSDet's performance in detecting new malware variants. For this purpose, we gathered data from a variety of sources (the Zoo dataset¹, from VirusTotal²) and from various portable benign tools such as Google play. We have gathered 5 540 Android apps where 3 440 are malicious and 2 100 apps are benign files. From those apps, a total of malicious motifs and a total of benign motifs were extracted. The conducted process is summarized in Table II. The Drebin dataset [31], which contains 123 453 benign applications and 5 560 malware samples, is used for the evaluation of our approach against the new variants of malware and 0-day attacks. The necessity for confirmation that ProRSDet is not fitting the base of examples led to the selection of a dataset that is different from the one used for the construction phase. For comparisons, various state-of-the-art methods were investigated. These are the classical classifiers named in Table V, tested using Weka with the proposed default parameters settings, three known methods (Rathore et al. [32], Gym-plus [33], and AMD [24]), and several commercial antimalware named in Table VII. The comparison made with the two recent state-of-the-art methods ([32] and [33]) is justified by the fact that those approaches are somehow similar to ProRSDet. In fact, there are common traits between our developed approach and those approaches: they propose a two-task solution (a malware generation task and a malware detection task). Also, to ensure the fairness of comparisons between evolutionary approaches (AMD [24] and ProRSDet), we used the parameter settings described in Table III.

Both evolutionary approaches perform 798 000 function evaluations in each run. Also, to help determine the most

TABLE II
NUMBER OF OBTAINED MOTIFS.

	Number of apps	Number of motifs
Benign	2 100	28 019 663
Malicious	3 440	36 995 382

TABLE III
EVOLUTIONARY PARAMETERS.

	ProRSDet	AMD
Population size	(both levels) 30	180
Generation size	(both levels) 30	4500
Mutation rate	0.5	0.5
Crossover rate	0.9	0.9

appropriate α and β values, a set of experiments is conducted and the results are reported in Table IV. Indeed, Table IV shows that the best results were reached with a pair of α and β value that equals 0.5, respectively. When running the experiments, we concluded that the fitness functions become stabilized around the 36th generation. For these reasons, the algorithms did not suffer from premature convergence. The metrics used for the evaluation are: true positives (TP), false positives (FP), true negatives (TN), false negatives (FN), recall (RC), specificity (SP), accuracy (AC), precision (PR), F1_score (FS), and the Area Under the Receiver Operating Characteristics (ROC) Curve (AUC). All of the conducted experiments, based on a 10-fold cross validation, are run on an Intel[®] Xeon[®] Processor CPU E5-2620 v3, with a 16 GB RAM.

B. Results Analysis

In this section, we compare the ProRSDet obtained results to a set of classifiers (Table V), three state-of-the-art approaches (Rathore et al. [32], Gym-plus [33] and AMD [24]) and five antivirus engines (Table VI and Table VII). More precisely, to determine how accurate our predictive model will perform in practice, we used 10-fold cross-validation. We considered all of the collected programs and hence all of the obtained motifs stored in *SBM* and *SMM* (see Table II). So, concerning the comparison with the top-five classifiers (Table V), and based on all of the evaluation metrics, ProRSDet surpasses all other classifiers. In comparison to the *LDA* and *J48* classifiers, which produced the second best results among the rest of the classifiers with a pair of precision and accuracy of (98.36%, 97.82%) for *LDA* and (97.73%, 96.58%) for *J48* and a pair of F1_score and specificity of (97.32%, 97.31%) for *LDA* and (98.37%, 97.13%) for *J48*, ProRSDet achieved a precision of 98.20%, an accuracy of 98.22%, an F1 score of 98.21%, and a specificity of 98.20%. These remarkable ProRSDet results are based on its high true positives (98.20%) and low false positives (1.80%), which are the best achieved values among the results of the classifiers. These encouraging findings show that ProRSDet is capable of distinguishing between the two possible designations (malicious and benign). Also, regarding the comparison between the EA-based approaches (ProRSDet and AMD [24]), we used an unknown dataset (Drebin dataset

¹<https://thezoo.morirt.com/>

²<https://www.virustotal.com/gui/home/upload>

²<https://www.cs.waikato.ac.nz/ml/weka/>

TABLE IV
VPRS PARAMETERS.

		RC	SP	AC	PR	FS	AUC	FPR	FNR
Experiment 1	$\alpha = 1$ $\beta = 0$	97.56	97.02	97.29	97.01	97.28	80.01	02.99	02.42
Experiment 2	$\alpha = 0.85$ $\beta = 0.15$	96.68	97.66	97.17	97.69	97.18	82.13	02.31	03.28
Experiment 3	$\alpha = 0.7$ $\beta = 0.3$	96.66	97.07	96.87	97.09	96.87	73.80	02.91	03.35
Experiment 4	$\alpha = 0.5$ $\beta = 0.5$	97.99	97.32	97.66	97.31	96.65	87.00	02.69	01.99

[31]) and ProRSDet outperformed AMD in terms of the used evaluation metrics as stated in Table VI. This can be explained by the contribution brought by the VPRS Analyzer which helped keep the most "relevant" malicious motifs.

Moreover, we may derive from Table VI and Table VII that, when compared to competing state-of-the-art approaches, (Rathore et al. [32], Gym-plus [33] and AMD [24]) using the unknown dataset [31], ProRSDet came in top with an accuracy of 97.66%, a specificity of 97.32%, a recall of 97.99%, a precision of 97.31%, and an AUC of 86.15%. Rathore et al., Gym-plus and AMD, obtained an accuracy of 93.81%, 93.50% and 92.28%, respectively, which are lower than those obtained by our proposed technique. In addition, the interesting detection results obtained by ProRSDet are endorsed by the results presented in Table VII which refers to the comparison with the commercial antivirus engines. Table VII shows that ProRSDet reached an accuracy rate of 97.66% whereas the ESET NOD32 engine, which is ranked first among all the other malware antivirus engines, registered only 66.68% of accuracy. It is to be noted that the accuracy values of the four remaining antivirus engines varied approximately between 56% and 66%.

The results reported from Tables V, VI and VII highlight the ability of ProRSDet – thanks to its set of efficient produced rules which are generated using the most relevant set of the generated fraudulent malware; both guaranteed via the use of the BLOP architecture and the VPRS component – to achieve accurate detection operations against new and unknown variants of malware.

To better clarify the efficiency and benefits of relying on the bilevel architecture within ProRSDet, we analyse the results in terms of false positive and the false negative rates. The registered ProRSDet values of those two metrics (Table VI) confirm the usefulness of a bilevel architecture to detect a malicious code efficiently. The continuous competition between both levels (first layer and second layer) permitted generation of good solutions (detection rules and fraudulent motifs) and this had positive impact on the values of FPR (02.69%) and FNR (01.99%). In comparison to ProRSDet, the registered FPR and FNR values for AMD [24], which rely on a single-layer based architecture via the use of evolutionary algorithms, are 06.37% and 08.84%, respectively. In addition, referring to Table VIII, we can state that the Variable Precision Rough Set based module succeeded to set apart 198 522 ambiguous instances (*possible* set) among the generated fraudulent motifs *SHM*

(468 000 instances). More precisely, a set of 92 689 of false motifs were removed from the whole set of ambiguous motifs. The removal of those false motifs was performed thanks to the *Relevance_Malicious_Index* and after being processed by the lower and upper limit certainty thresholds explained and illustrated in Section IV-2. This distinction brings to light the VPRS component's important contribution in improving the quality of the fraudulent motifs by the second layer and which, consequently, positively affected the false alarms rate. Let us recall that ProRSDet provides a set of robust detection rules thanks to the set of malicious motifs produced by the second layer's GA reinforced by the VPRS module. The VPRS module helps determine the set of "certain" malicious motifs and the set of "possible" malicious motifs. When dealing with this set of *possible* motifs, a *Relevance_Malicious_index* (Equation 10) that estimates the reliability degree of each *possible* malicious motif is also provided to the user (as presented in Section IV-2). This metric, specific to the evaluation of each *possible* malicious motif, will help determine the fate of this specific motif: add it to the set of malicious motifs or remove it. To be more specific, Figure 4 represents the number of the obtained *possible malicious motifs* with regards to their *Relevance_Malicious_index*. Figure 4 shows that 51.28% of possible malicious motifs have a *Relevance_Malicious_index* that exceeds 50%. Also, 17.98% of possible rules succeeded to correctly classify apps with rates that lie between 41% and 50%. 18.23% of those rules ranked just below with a *Relevance_Malicious_index* comprised between 31% and 40%. Approximately only 13% of the rules failed to have a *Relevance_Malicious_index* above 30%. An example of the use of this index was previously given in Section IV-2. Also, an important aspect in our proposed ProRSDet approach that needs to be clarified concerns the setting of the α and β VPRS parameters. In fact, in this study we adopted the the trial and error method which consists of choosing randomly values and apply them in our algorithm. For instance, we conducted four different experiments (Table IV) which helped us determine the best pair of α and β values that leads to better detection results. More precisely, we tried different combination of α and β values and each time we assessed the recall (*RC*), specificity (*SP*), accuracy (*AC*), precision (*PR*), F1_score (*FS*), Area under ROC curve (*AUC*), false positive rate (*FPR*) and false negative rate (*FNR*). The best values were reached with $\alpha = 0.5$ and $\beta =$

TABLE V
COMPARISON BETWEEN PRORSDET AND THE CLASSICAL CLASSIFIERS.

Classifier/ approach	TP	FP	TN	FN	RC	SP	AC	PR	FS	AUC	FPR	FNR
ProRSDet	98.20	01.80	98.24	01.76	98.23	98.20	98.22	98.20	98.21	86.79	01.80	01.76
LR	93.81	06.19	96.75	03.25	96.65	93.98	95.28	93.17	95.60	63.69	06.01	03.34
NB	92.30	07.70	28.41	71.59	56.31	78.67	60.35	92.37	93.62	65.06	02.13	09.03
RF	97.41	02.59	95.90	04.10	96.00	98.37	97.16	97.36	97.17	73.04	02.62	04.03
J48	97.18	02.82	93.98	06.02	94.27	97.13	96.58	97.73	98.37	83.90	02.91	05.83
k-NN	89.52	10.48	95.21	04.79	94.92	90.08	92.37	85.74	90.56	57.69	09.91	05.07
LDA	97.29	02.71	98.36	01.64	98.34	97.31	97.82	98.36	97.32	75.96	02.68	01.65

LR: Logistic Regression; LDA: Linear Discriminant Analysis; RF: Random Forest; J48: Decision Tree; NB: Naive Bayes; k-NN: k-Nearest Neighbours.

TABLE VI
COMPARISON BETWEEN PRORSDET AND AMD [24] USING THE DREBIN DATASET [31].

Classifier/ approach	TP	FP	TN	FN	RC	SP	AC	PR	FS	AUC	FPR	FNR
ProRSDet	97.31	02.69	98.01	01.99	97.99	97.32	97.66	97.31	96.65	86.15	02.69	01.99
AMD	93.80	06.19	90.90	09.10	96.20	92.70	92.28	93.60	92.37	57.69	06.37	08.84

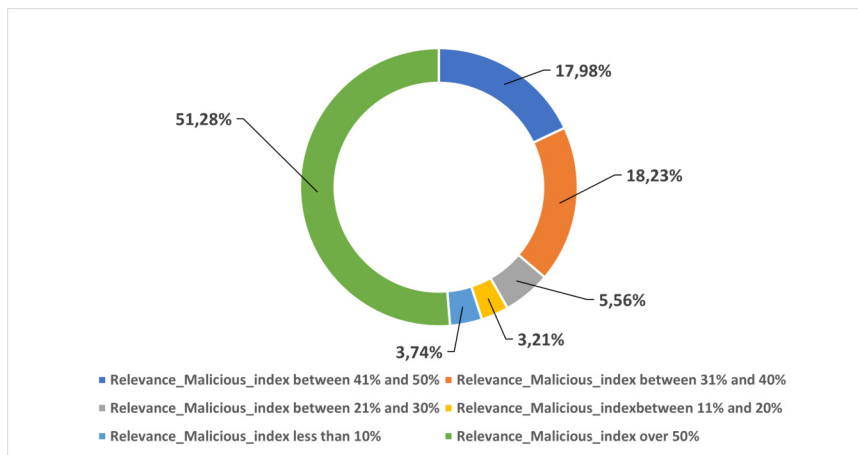


Fig. 4. Number of possible motifs with regards to the *Relevance_Malicious_index*.

TABLE VII
ACCURACY RESULTS OF PRORSDET AND TOP FIVE COMMERCIAL ENGINES BY VIRUSTOTAL³ ON THE DREBIN DATASET [31].

Anti-malware	Reference	Accuracy (%)
ProRSDet	Our current approach	97.66
ESET NOD32	https://www.eset.com	66.68
AegisLab	www.aegislab.com	66.23
NANO antivirus	http://www.nanoav.ru	66.23
VIPRE	https://www.vipre.com	62.53
McAfee	https://www.mcafee.com	56.21
Gym-plus	[33]	93.50
Rathore et al.	[32]	93.81 (with RF)

TABLE VIII
NUMBERS OF RELEVANT AND AMBIGUOUS GENERATED MALICIOUS MOTIFS.

Number of generated motifs in ProRSDet		
Possible instances		Certain instances
False motifs	Approved motifs	
92 689	105 833	269 478

VI. CONCLUSION AND FUTURE DIRECTIONS

In this research, we developed ProRSDet, a malware detection technique that combines the Variable Precision Rough Set model and bilevel optimization. Within the bilevel architecture, the malware generation task (inner algorithm or second layer) and the rules generation task (detection task, outer algorithm or first layer) are in mutual competition. The second layer generates “Relevant” malicious motifs which are generated by a GA and thoroughly checked by a VPRS component that only keeps the most “pertinent” ones, and which are capable of eluding the GP’s set of detection rules in the first layer.

0.5. Indeed, we specifically registered an *AC* of 97.66%, a *PR* of 97.31% a *FPR* of 02.69% and a *FNR* of 01.99%. Those significant values found their way thanks to great reached percentages of true positives and true negatives.

These effective created detection rules, in turn, attempt their hardest to detect the second layer's set of fraudulent patterns.

ProRSDet outperformed a variety of state-of-the-art approaches and commercial engines, achieving encouraging detection rates of 97.66% accuracy and 2.69% false positives. We plan to investigate other methods to help determine the best values of α and β concerning the VPRS. It would be interesting to design an adaptive parameter tuning strategy that aims to approximate the best values of the VPRS parameters. Also, we can consider other theories that deal with the inconsistency in the future (i.e., [34], [35]), as well as consider expanding the scope of the proposed work to encompass other operating systems.

REFERENCES

- [1] G. Ollmann, "The evolution of commercial malware development kits and colour-by-numbers custom malware," *Computer Fraud & Security*, vol. 2008, no. 9, pp. 4–7, 2008. doi: 10.1016/S1361-3723(08)70135-0
- [2] D. Li, T. Qiu, S. Chen, Q. Li, and S. Xu, "Can we leverage predictive uncertainty to detect dataset shift and adversarial examples in android malware detection?" in *Annual Computer Security Applications Conference*, 2021. doi: <https://doi.org/10.1145/3485832.3485916> pp. 596–608.
- [3] I. Santos, J. Nieves, and P. G. Bringas, "Semi-supervised learning for unknown malware detection," in *International Symposium on Distributed Computing and Artificial Intelligence*. Springer, 2011. doi: 10.1007/978-3-642-19934-9_53 pp. 415–422.
- [4] M. Nauman, N. Azam, and J. Yao, "A three-way decision making approach to malware analysis using probabilistic rough sets," *Information Sciences*, vol. 374, pp. 193–209, 2016. doi: <https://doi.org/10.1016/j.ins.2016.09.037>
- [5] K. Riad and L. Ke, "Roughdroid: operative scheme for functional android malware detection," *Security and Communication Networks*, vol. 2018, 2018. doi: <https://doi.org/10.1155/2018/8087303>
- [6] S. Piparia, D. Adamo, R. Bryce, H. Do, and B. Bryant, "Combinatorial testing of context aware android applications," in *2021 16th Conference on Computer Science and Intelligence Systems (FedCSIS)*, 2021. doi: 10.15439/2021F003 pp. 17–26.
- [7] F. Alotaibi and A. Lisitsa, "Matrix profile for ddos attacks detection," in *2021 16th Conference on Computer Science and Intelligence Systems (FedCSIS)*, 2021. doi: 10.15439/2021F114 pp. 357–361.
- [8] P. Kishore, S. K. Barisal, and D. P. Mohapatra, "Decentralized controller for software interconnected system subject to malicious attacks," in *FedCSIS (Position Papers)*, 2021. doi: 10.15439/2021F90 pp. 211–218.
- [9] K. Xu, Y. Li, R. Deng, K. Chen, and J. Xu, "Droidevolver: Self-evolving android malware detection system," in *2019 IEEE European Symposium on Security and Privacy (EuroSP)*, 2019. doi: 10.1109/EuroSP.2019.00014 pp. 47–62.
- [10] F. Cara, M. Scalas, G. Giacinto, and D. Maiorca, "On the feasibility of adversarial sample creation using the android system api," *Information*, vol. 11, no. 9, p. 433, 2020. doi: <https://doi.org/10.3390/info11090433>
- [11] W. Hu and Y. Tan, "Generating adversarial malware examples for black-box attacks based on gan," *arXiv preprint arXiv:1702.05983*, 2017. doi: <https://doi.org/10.48550/arXiv.1702.05983>
- [12] Z. Moti, S. Hashemi, and A. Namavar, "Discovering future malware variants by generating new malware samples using generative adversarial network," in *2019 9th International Conference on Computer and Knowledge Engineering (ICCKE)*, 2019. doi: 10.1109/ICCKE48569.2019.8964913 pp. 319–324.
- [13] V. Yegneswaran, J. T. Giffin, P. Barford, and S. Jha, "An architecture for generating semantic aware signatures," in *USENIX security symposium*, 2005, pp. 97–112.
- [14] S. Singh, C. Estan, G. Varghese, and S. Savage, "Automated worm fingerprinting," in *OSDI*, vol. 4, 2004, pp. 4–4.
- [15] K. Griffin, S. Schneider, X. Hu, and T.-c. Chiueh, "Automatic generation of string signatures for malware detection," in *International workshop on recent advances in intrusion detection*. Springer, 2009. doi: https://doi.org/10.1007/978-3-642-04342-0_6 pp. 101–120.
- [16] J. O. Kephart, "Automatic extraction of computer virus signatures," in *Proc. 4th Virus Bulletin International Conference, Abingdon, England, 1994*, 1994, pp. 178–184.
- [17] Z. Li, M. Sanghi, Y. Chen, M.-Y. Kao, and B. Chavez, "Hamsa: fast signature generation for zero-day polymorphic worms with provable attack resilience," in *2006 IEEE Symposium on Security and Privacy (S P'06)*, 2006. doi: 10.1109/SP.2006.18 pp. 15 pp.–47.
- [18] J. Newsome, B. Karp, and D. Song, "Polygraph: automatically generating signatures for polymorphic worms," in *2005 IEEE Symposium on Security and Privacy (S P'05)*, 2005. doi: 10.1109/SP.2005.15 pp. 226–241.
- [19] E. Aydogan and S. Sen, "Automatic generation of mobile malwares using genetic programming," in *European conference on the applications of evolutionary computation*. Springer, 2015. doi: 10.1007/978-3-319-16549-3_60 pp. 745–756.
- [20] M. F. Zolkipli and A. Jantan, "A framework for malware detection using combination technique and signature generation," in *2010 Second International Conference on Computer Research and Development*. IEEE, 2010. doi: 10.1109/ICCRD.2010.25 pp. 196–199.
- [21] H. G. Kayacık, A. N. Zincir-Heywood, and M. I. Heywood, "Can a good offense be a good defense? vulnerability testing of anomaly detectors through an artificial arms race," *Applied Soft Computing*, vol. 11, no. 7, pp. 4366–4383, 2011. doi: <https://doi.org/10.1016/j.asoc.2010.09.005>
- [22] Y. Xue, G. Meng, Y. Liu, T. H. Tan, H. Chen, J. Sun, and J. Zhang, "Auditing anti-malware tools by evolving android malware and dynamic loading technique," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 7, pp. 1529–1544, 2017. doi: 10.1109/TIFS.2017.2661723
- [23] S. Sen, E. Aydogan, and A. I. Aysan, "Coevolution of mobile malware and anti-malware," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 10, pp. 2563–2574, 2018. doi: 10.1109/TIFS.2018.2824250
- [24] M. Jerbi, Z. C. Dagdia, S. Bechikh, M. Makhlof, and L. B. Said, "On the use of artificial malicious patterns for android malware detection," *Computers & Security*, p. 101743, 2020. doi: <https://doi.org/10.1016/j.cose.2020.101743>
- [25] M. Jerbi, Z. C. Dagdia, S. Bechikh, and L. B. Said, "Android malware detection as a bi-level problem," *Computers & Security*, p. 102825, 2022. doi: <https://doi.org/10.1016/j.cose.2022.102825>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S016740482200219X>
- [26] M. Jerbi, Z. Chelly Dagdia, S. Bechikh, and L. Ben Said, "Malware detection using rough set based evolutionary optimization," in *Neural Information Processing*, T. Mantoro, M. Lee, M. A. Ayu, K. W. Wong, and A. N. Hidayanto, Eds. Cham: Springer International Publishing, 2021. ISBN 978-3-030-92307-5 pp. 634–641.
- [27] B. Colson, P. Marcotte, and G. Savard, "An overview of bilevel optimization," *Annals of operations research*, vol. 153, no. 1, pp. 235–256, 2007. doi: <https://doi.org/10.1007/s10479-007-0176-2>
- [28] J.-A. Mejía-de Dios, E. Mezura-Montes, and M. Quiroz, "Automated parameter tuning as a bilevel optimization problem solved by a surrogate-assisted population-based approach," *Applied Intelligence*, vol. 51, pp. 1–23, 08 2021. doi: 10.1007/s10489-020-02151-y
- [29] W. Ziarko, "Set approximation quality measures in the variable precision rough set model," in *HIS*, 2002, pp. 442–452.
- [30] L. Nanni and A. Lumini, "Generalized needleman-wunsch algorithm for the recognition of t-cell epitopes," *Expert Systems with Applications*, vol. 35, no. 3, pp. 1463–1467, 2008. doi: <https://doi.org/10.1016/j.eswa.2007.08.028>
- [31] D. Arp, M. Spreitzenbarth, M. Hubner, H. Gascon, K. Rieck, and C. Siemens, "Drebin: Effective and explainable detection of android malware in your pocket," in *Ndss*, vol. 14, 2014. doi: 10.14722/ndss.2014.23247 pp. 23–26.
- [32] H. Rathore, S. K. Sahay, P. Nikam, and M. Sewak, "Robust android malware detection system against adversarial attacks using q-learning," *Information Systems Frontiers*, vol. 23, no. 4, pp. 867–882, 2021. doi: <https://doi.org/10.1007/s10796-020-10083-8>
- [33] C. Wu, J. Shi, Y. Yang, and W. Li, "Enhancing machine learning based malware detection model by reinforcement learning," in *Proceedings of the 8th International Conference on Communication and Network Security*, 2018. doi: <https://doi.org/10.1145/3290480.3290494> pp. 74–78.
- [34] D. Ślęzak, "Rough sets and bayes factor," in *Transactions on Rough Sets III*. Springer, 2005, pp. 202–229.
- [35] D. Slezak and W. Ziarko, "The investigation of the bayesian rough set model," *International journal of approximate reasoning*, vol. 40, no. 1-2, pp. 81–91, 2005. doi: <https://doi.org/10.1016/j.ijar.2004.11.004>

Multi-Criteria Decision-Making with Linguistic Labels

Alicja Mieszkowicz-Rolka and Leszek Rolka
Rzeszów University of Technology
Al. Powstańców Warszawy 8, 35-959 Rzeszów, Poland
Email: {alicjamr, leszokr}@prz.edu.pl

Abstract—This paper proposes an approach that is suitable for solving multi-criteria decision-making problems characterized by fuzzy (subjective) criteria. A finite set (universe) of alternatives will be expressed as a decision table that represents a fuzzy information system in which every fuzzy criterion is connected with a set of its linguistic values. We apply subjective preference degrees for linguistic values that should be provided by a decision-maker. To simplify the process of decision-making in big data environments, an additional stage will be introduced that can produce a smaller set of alternatives represented by fuzzy linguistic labels of similarity classes. We select a small set of similarity classes for the final ranking. A measure of compatibility will be defined that should express the accordance of a selected alternative with preferences given for the linguistic values of a particular fuzzy criterion.

I. INTRODUCTION

MULTI-CRITERIA decision-making is a very important task that has to be performed in various areas of human activity, especially in technique, industry, economy, business, and in everyday life. Depending on the number of considered criteria and the size of the solution space (number of alternatives), the process of determining the best alternative can be problematic, therefore, only small-sized and relative simple decision-making tasks can be effectively solved by humans. Moreover, as a general rule, there is no accordance in monotonicity between particular criteria, i.e., one obtains different rankings of alternatives for each criterion, hence a compromise solution should be determined. In recent decades, several approaches were proposed for solving multi-criteria decision-making problems, e.g., AHP, TOPSIS, VIKOR, PROMETHE, and ELECTRE [1]. In the case of very large solution spaces, for example when dealing with the combinatorial optimization problems, metaheuristic approaches such as simulated annealing, swarm optimization, and genetic algorithms can be applied.

Another issue, often encountered in practical optimization tasks, is uncertainty and vagueness in the characteristic of evaluated alternatives. Standard optimization methods only base on classic knowledge representation by utilizing deterministic objective functions and constraints, crisp set theory, and bi-valued logic. In order to take vagueness and subjectivity into account, many decision-making and optimization algorithms have been extended or combined with various soft computing methods in the form of hybrid approaches.

The fuzzy set theory has proved to be one of the most successful paradigms for dealing with problems that are vague in nature and require the use of linguistic terms or subjective evaluation. This makes it possible to use notions such as “quite large” or “very young”. Linguistic values expressed as fuzzy sets were widely applied and adopted by many researchers, who introduced several generalization, e.g., intuitionistic, type-2, hesitant fuzzy sets, and hybrid fuzzy-rough set models [2], [3].

Because in many decision-making algorithms a fuzzy representation of knowledge could be successfully implemented [4], [5], popular multi-criteria optimization algorithms have also been extended to deal with fuzzy terms instead of crisp numbers [6], [7].

In our previous work [8], we introduced the concept of fuzzy linguistic label that can be utilized in the framework of fuzzy or fuzzy-rough decision systems. The principle of the label-based approach consists in a simplified way of comparing and classifying the elements of a universe that are described with fuzzy attributes. Such an approach can be helpful in analysis of complex information systems that have large number of attributes and fuzzy linguistic values. The obtained results strongly depend on the used fuzzy operators and they may be not easy to interpret. This is why the label-based approach does not use standard fuzzy relation for determining classes of similar elements (alternatives) of the universe. Instead, by finding positive (dominant) linguistic values of attributes (criteria), a common description of similar objects can be easily obtained in the form of fuzzy linguistic labels.

In the current work, we propose a label-based approach to multi-criteria decision-making. We introduce several new ideas. First, we extend the requirements imposed on the membership degree of elements in linguistic values of attributes in the information system. This can be helpful in the process of preparing a consistent high-quality decision table by an expert. As only two neighbouring linguistic values of criteria can be activated, the decision table can be given in a compact form. Secondly, we propose a method of finding the best alternatives in the case of subjective fuzzy criteria. We define compatibility measure that can be used to evaluate the accordance of alternatives with the subjective preferences for linguistic values that are required by a decision-maker. Furthermore, the approach consists of two stages. This can be useful when the process of decision-making is performed

in a big-data environment. We avoid detailed evaluation of each alternative in a huge solution space. Instead, the best linguistic labels can be easily discovered at the first stage. The final solution can then be obtained by examining only a small number of the promising similarity classes of alternatives connected with the selected linguistic labels.

II. FUZZY DECISION SYSTEMS

To formalize the process of multi-criteria decision-making in a fuzzy environment, we should use the notion of fuzzy information system [9] that is expressed as a 4-tuple

$$\text{FDS} = \langle U, A, \mathbb{V}, f \rangle, \quad (1)$$

where:

U – is a nonempty set (universe) of elements (alternatives),

A – is a finite set of fuzzy attributes (criteria),

\mathbb{V} – is a set of linguistic values of criteria, $\mathbb{V} = \bigcup_{a \in A} \mathbb{V}_a$,
 \mathbb{V}_a is the set of linguistic values of a criterion $a \in A$,

f – is an information function, $f : U \times \mathbb{V} \rightarrow [0, 1]$,
 $f(x, V) \in [0, 1]$, for all $x \in U$, and $V \in \mathbb{V}$.

Any fuzzy criterion $a_i \in A$, where $i = 1, 2, \dots, n$, can take value from the family of its linguistic values denoted by $\mathbb{A}_i = \{A_{i1}, A_{i2}, \dots, A_{in_i}\}$. For all elements $x \in U$, the membership degree in the linguistic values of all fuzzy criteria will be assigned by experts. We require the following conditions to be satisfied, when the membership is assigned:

$$\begin{aligned} \exists A_{ik} \in \mathbb{A}_i \quad & (\mu_{A_{ik}}(x) \geq 0.5, \\ & \mu_{A_{ik-1}}(x) = 1 - \mu_{A_{ik}}(x) \vee \\ & \mu_{A_{ik+1}}(x) = 1 - \mu_{A_{ik}}(x)), \end{aligned} \quad (2)$$

$$\text{power}(\mathbb{A}_i(x)) = \sum_{k=1}^{n_i} \mu_{A_{ik}}(x) = 1. \quad (3)$$

The requirements (2), and (3) constitute a generalization of the properties that can be observed in crisp information systems. They are necessary for creating well-defined and consistent information systems.

Every information system can be represented by a decision table in which the rows correspond to the elements of the universe U , and the columns to the linguistic values of criteria. Due to the requirement (2), we would obtain a large sparse decision matrix, when the number of criteria and the number of their linguistic values is large. Therefore, it is more convenient to introduce a compact form of a decision table that only contains the information about the dominant linguistic values, which satisfy the requirement (2), for every element of the universe U .

Membership degree of any element $x \in U$, in the dominant k -th linguistic value of a selected fuzzy criterion $a_i \in A$, will be expressed in the following form

$$\mu_{A_{ik}}(x)(\text{neighbour})/A_{ik} \quad (4)$$

where:

$\text{neighbour} = L$ indicates a nonzero membership degree in the neighbouring left linguistic value, with $\mu_{A_{ik-1}}(x) = 1 - \mu_{A_{ik}}(x)$,

$\text{neighbour} = R$ indicates a nonzero membership degree in the neighbouring right linguistic value, with $\mu_{A_{ik+1}}(x) = 1 - \mu_{A_{ik}}(x)$,

$\text{neighbour} = C$ indicates no membership in the right and left neighbouring linguistic values.

The introduced concepts are used in an illustrative example in Section V.

III. LINGUISTIC LABELS

The elements of a given universe of discourse can be compared to each other by using a binary relation for determining the degree of their similarity. In the standard rough set theory [10], a crisp indiscernibility relation is utilized that generates a partition of the universe into classes of elements that cannot be discerned, because they have the same value of (selected) attributes. Comparison of elements in a fuzzy model can be done in different ways, because various fuzzy operators can be used to determine the similarity of elements [2]. Another difficulty arises in analysis of big information systems that have not only large number of elements, but also many fuzzy attributes and linguistic values. In consequence, one can obtain a vast number of similarity classes, which complicates the calculations and, above all, makes it difficult to interpret the results. This issue was an inspiration for proposing a straightforward method of classifying the elements of fuzzy decision systems [8].

The principle of this approach consists in finding classes of characteristic elements of the universe that have the same description given in the form of tuple of dominant linguistic values of attributes. According to the requirement (2), for every element of the universe U , and every attribute $a \in A$, a distinct linguistic value can be found for which the membership degree has the greatest value. Hence, we do not apply a standard fuzzy similarity relation, but only identify dominant linguistic values in all rows of the decision table. This way we can easily discover the groups of (characteristic) elements that have the same dominant linguistic values of all attributes.

A tuple of dominant linguistic values of attributes is called a linguistic label. We can say that the characteristic elements of a linguistic label belong to the same similarity class. The degrees of membership in the dominant linguistic values are in a general case different numbers from the interval $[0.5, 1]$ for every element of the similarity class. However, we can also introduce an ideal element with the membership degree equal to 1 for all dominant linguistic values. Such ideal elements can be seen as an abstract representation of the linguistic labels.

By using linguistic labels, one is able to imitate the process of classifying objects that can be observed in human experts. Instead of performing an exhaustive comparison for every pair of elements of the universe, they rather try to discover a limited subset of ideal elements having a common characteristic.

Now, let us recall the basic notions of the label-based approach. The characteristic elements of the universe will be determined directly from the decision table, by respecting their membership in the linguistic values of all fuzzy attributes. Since the dominance of a linguistic value is a matter of the membership degree, we need to use a threshold of similarity, denoted by β , which satisfies the inequality

$$0.5 < \beta \leq 1. \quad (5)$$

A suitable value of the parameter β should be chosen as the threshold of similarity for classifying linguistic values of attributes.

Given a fuzzy information system FDS, we define [8] for any element $x \in U$, and any fuzzy attribute $a \in A$: the set $\widehat{\mathbb{V}}_a(x) \subseteq \mathbb{V}_a$ of positive linguistic values

$$\widehat{\mathbb{V}}_a(x) = \{V \in \mathbb{V}_a : f(x, V) \geq \beta\}, \quad (6)$$

the set $\overline{\mathbb{V}}_a(x) \subseteq \mathbb{V}_a$ of boundary linguistic values

$$\overline{\mathbb{V}}_a(x) = \{V \in \mathbb{V}_a : 0.5 \leq f(x, V) < \beta\}, \quad (7)$$

and the set $\check{\mathbb{V}}_a(x) \subseteq \mathbb{V}_a$ of negative linguistic values

$$\check{\mathbb{V}}_a(x) = \{V \in \mathbb{V}_a : 0 \leq f(x, V) < 0.5\}. \quad (8)$$

Due to the constraints (2) and (3), the sets $\widehat{\mathbb{V}}_a(x)$, $\overline{\mathbb{V}}_a(x)$, and $\check{\mathbb{V}}_a(x)$ have the following properties [8]:

$$(P1) \quad \text{card}(\widehat{\mathbb{V}}_a(x)) \leq 1,$$

$$(P2) \quad \text{card}(\overline{\mathbb{V}}_a(x)) \leq 2,$$

$$(P3) \quad \text{card}(\check{\mathbb{V}}_a(x)) < |\mathbb{V}_a|.$$

Every element $x \in U$ can be described with a combination of those linguistic values that are positive for that particular element. In this way, we determine the linguistic labels for all elements of the universe.

Formally, the set of linguistic labels $\widehat{\mathbb{L}}(x)$ is equal to the Cartesian product of the sets of positive linguistic values $\widehat{\mathbb{V}}_a(x)$, for all $a \in A$:

$$\widehat{\mathbb{L}}(x) = \prod_{a \in A} \widehat{\mathbb{V}}_a(x). \quad (9)$$

When inspecting the decision table, we can also discover elements $x \in U$ which have a common linguistic label $L(x)$.

By X_L , we denote the subset of the elements $x \in U$ that correspond to a linguistic label $L \in \mathbb{L}$, for all fuzzy attributes $a \in A$:

$$X_L = \{x \in U : L(x) = L\}. \quad (10)$$

The subset X_L is called the set of characteristic elements of the linguistic label L .

A linguistic label $L \in \mathbb{L}$ can be represented by an ordered tuple of positive linguistic values, for all attributes $a \in A$:

$$L = (\widehat{V}_{a_1}^L, \widehat{V}_{a_2}^L, \dots, \widehat{V}_{a_n}^L). \quad (11)$$

In the present paper, the notion of an attribute denotes a criterion, and an element of the universe is called an alternative.

IV. LABEL-BASED EVALUATION OF ALTERNATIVES

We divide the process of searching for the best alternatives into two stages. First, all rows of the decision table have to be inspected for determining the linguistic labels which are present in the information system.

A. Stage I

At the first stage, every linguistic label will be evaluated by determining its accordance with the preferences for linguistic values provided by a decision-maker. We also take into account the weights of criteria. The decision-maker should give the vector of weights $W = [w_1, w_2, \dots, w_n]$, which usually satisfy the requirement: $\sum_{i=1}^n w_i = 1$.

Let us denote by $pref(V)$ the preference for the linguistic value $V \in \mathbb{V}$. The compatibility of a linguistic label with the preferences for linguistic values of criteria can be presented in a detailed manner as a fuzzy set C_L on the domain of positive linguistic values of the label L

$$C_L = \{pref(\widehat{V}_{a_1}^L)/\widehat{V}_{a_1}^L, pref(\widehat{V}_{a_2}^L)/\widehat{V}_{a_2}^L, \dots, pref(\widehat{V}_{a_n}^L)/\widehat{V}_{a_n}^L\}. \quad (12)$$

Now, we define the measure of weighted compatibility of the linguistic label L with the preferences of the decision-maker as follows

$$compat(L) = \sum_{i=1}^n w_i \cdot pref(\widehat{V}_{a_i}^L). \quad (13)$$

where $\widehat{V}_{a_i}^L$ denotes the positive linguistic value of the criterion a_i in the linguistic label L , as given in the formula (11).

By applying the measure (13), a ranking of all linguistic labels can be determined. Basing on the ranking of the linguistic labels, we can select a group of the best linguistic labels and their characteristic elements as the promising candidates for generating the set of the best alternatives.

B. Stage II

At the second stage, evaluation of alternatives from the classes of characteristic elements of the selected linguistic labels is performed. The analyzed alternatives are represented in the form (4).

We define the measure of the weighted compatibility of an alternative $x \in U$ with the preferences for the linguistic values of attributes as follows

$$compat(x) = \sum_{i=1}^n w_i \cdot pref(V(x, i)) \cdot \mu_{V(x, i)}(x) + \sum_{i=1}^n w_i \cdot pref(N(x, i)) \cdot \mu_{N(x, i)}(x), \quad (14)$$

where:

$V(x, i)$ – is the positive linguistic value of the criterion a_i in the linguistic label $L(x)$, $V(x, i) = \widehat{V}_{a_i}^L(x)$,

$N(x, i)$ – is the neighbouring linguistic value of a_i in the linguistic label $L(x)$, $\mu_{N(x, i)}(x) = 1 - \mu_{V(x, i)}(x)$.

The measure (14) will be used to generate a set of the best alternatives.

V. EXAMPLE

Let us consider a fuzzy information system that includes alternatives: x_1, x_2, \dots, x_{15} . There are three fuzzy criteria: c_1, c_2 , and c_3 . The criteria c_1 and c_3 can take three linguistic values, whereas the criterion c_2 five linguistic values.

We assume that the decision regarding the degree of membership of every alternative in the linguistic values of all fuzzy criteria was made taking into account the requirements (2) and (3). The fuzzy information system was prepared in a compact form (Table I), according to the formula (4).

TABLE I
COMPACT DECISION TABLE

	a_1	a_2	a_3
x_1	0.8(L)/ A_{12}	0.7(L)/ A_{23}	0.7(L)/ A_{33}
x_2	0.8(L)/ A_{12}	0.7(L)/ A_{24}	0.8(L)/ A_{32}
x_3	0.8(R)/ A_{12}	0.7(R)/ A_{22}	0.8(L)/ A_{32}
x_4	0.8(R)/ A_{12}	0.7(R)/ A_{23}	0.7(L)/ A_{33}
x_5	0.8(R)/ A_{12}	0.7(R)/ A_{24}	0.8(R)/ A_{32}
x_6	0.8(L)/ A_{12}	0.7(L)/ A_{22}	0.8(R)/ A_{32}
x_7	0.8(L)/ A_{12}	0.7(R)/ A_{23}	0.7(L)/ A_{33}
x_8	0.7(R)/ A_{11}	0.7(L)/ A_{23}	0.6(L)/ A_{33}
x_9	0.7(L)/ A_{13}	0.8(L)/ A_{23}	0.7(L)/ A_{33}
x_{10}	0.7(L)/ A_{13}	0.8(R)/ A_{23}	0.7(L)/ A_{33}
x_{11}	0.8(L)/ A_{13}	0.8(L)/ A_{25}	0.7(L)/ A_{32}
x_{12}	0.8(R)/ A_{12}	0.7(L)/ A_{23}	0.7(L)/ A_{33}
x_{13}	0.8(L)/ A_{13}	0.8(L)/ A_{25}	0.7(R)/ A_{32}
x_{14}	0.8(R)/ A_{11}	0.7(R)/ A_{23}	0.8(L)/ A_{33}
x_{15}	0.8(R)/ A_{11}	0.7(R)/ A_{21}	1.0(C)/ A_{33}

We also present the full decision table (Table II) with emphasized values of membership degree for the linguistic values which are dominant.

To assess mainly the influence of preferences of linguistic values, we can take the same weight for every criterion by using the following vector of weights: $W = [0.33, 0.33, 0.34]$. The preferences for the linguistic values of all criteria are given in Table III.

By inspecting the decision table, we obtain the following linguistic labels with their characteristic elements:

$$\begin{aligned}
 L_1 &= (A_{12}A_{23}A_{33}) : X_{L_1} = \{x_1, x_4, x_7, x_{12}\}, \\
 L_2 &= (A_{11}A_{23}A_{33}) : X_{L_2} = \{x_8, x_{14}\}, \\
 L_3 &= (A_{12}A_{24}A_{32}) : X_{L_3} = \{x_2, x_5\}, \\
 L_4 &= (A_{12}A_{22}A_{32}) : X_{L_4} = \{x_3, x_6\}, \\
 L_5 &= (A_{13}A_{23}A_{33}) : X_{L_5} = \{x_9, x_{10}\}, \\
 L_6 &= (A_{13}A_{25}A_{32}) : X_{L_6} = \{x_{11}, x_{13}\}, \\
 L_7 &= (A_{11}A_{21}A_{33}) : X_{L_7} = \{x_{15}\}.
 \end{aligned}$$

A. Stage I

For all obtained linguistic labels, we determine the sets representing the compatibility with the preferences of the

decision-maker, according to the formula (12)

$$\begin{aligned}
 C_{L_1} &= \{0.50/A_{12}, 1.00/A_{23}, 1.00/A_{33}\}, \\
 C_{L_2} &= \{1.00/A_{11}, 1.00/A_{23}, 1.00/A_{33}\}, \\
 C_{L_3} &= \{0.50/A_{12}, 0.75/A_{24}, 0.50/A_{32}\}, \\
 C_{L_4} &= \{0.50/A_{12}, 0.50/A_{22}, 0.50/A_{32}\}, \\
 C_{L_5} &= \{0.25/A_{13}, 1.00/A_{23}, 1.00/A_{33}\}, \\
 C_{L_6} &= \{0.25/A_{13}, 0.25/A_{25}, 0.50/A_{32}\}, \\
 C_{L_7} &= \{1.00/A_{11}, 0.00/A_{21}, 1.00/A_{33}\}.
 \end{aligned}$$

Next, we determine the weighted compatibility of all linguistic labels according to the formula (13). The obtained ranking of the linguistic labels is included in Table IV.

B. Stage II

Those similarity classes of the linguistic labels that have the greatest compatibility with the preferences for linguistic values of criteria will be selected for a detailed analysis of their alternatives. In a real-world application with a huge number of alternatives, the worst similarity classes would be discarded from further consideration. In our small example, all linguistic labels are taken into account for determining the ranking of all alternatives.

We demonstrate the evaluation of the alternative x_8 that is a characteristic element of the linguistic label L_2 . From Table I, we take the entry 0.7(R)/ A_{11} expressing the fuzzy value of x_8 for the criterion a_1 . The alternative has a membership degree in the positive linguistic value A_{11} equal to 0.7, hence we have $V(x_8, 1)(x_8) = A_{11}$, and $\mu_{V(x_8, 1)}(x_8) = 0.7$. There is also a nonzero membership in the right neighbouring linguistic value A_{12} , that is, $N(x_8, 1)(x_8) = A_{12}$, and $\mu_{N(x_8, 1)}(x_8) = 1 - 0.7 = 0.3$.

By the same way, the terms for the attributes a_2 and a_3 can be obtained. Therefore, according to the formula (14), the value $compat(x_8)$ is equal to:

$$\begin{aligned}
 &w_1 \cdot (pref(A_{11}) \cdot \mu_{A_{11}}(x_8) + pref(A_{12}) \cdot \mu_{A_{12}}(x_8)) + \\
 &w_2 \cdot (pref(A_{23}) \cdot \mu_{A_{23}}(x_8) + pref(A_{22}) \cdot \mu_{A_{22}}(x_8)) + \\
 &w_3 \cdot (pref(A_{33}) \cdot \mu_{A_{33}}(x_8) + pref(A_{32}) \cdot \mu_{A_{32}}(x_8)) = 0.833
 \end{aligned}$$

The results obtained for all alternatives are presented in Table IV. We get the following ordering of alternatives: $x_{14}, x_8, x_7, x_1, x_4, x_{12}, x_{10}, x_9, x_2, x_5, x_6, x_3, x_{13}, x_{11}$. As we can see, the alternatives which are characteristic elements of the best linguistic labels have a high degree of compatibility with the preferences of the decision-maker.

VI. CONCLUSIONS

Linguistic labels can be effectively used in analysis of fuzzy information systems, including multi-criteria optimization tasks. The concept of fuzzy linguistic label was inspired by observation of the decision-making activity performed by human experts. The label-based approach presented in this paper takes into account subjective preferences for linguistic values of fuzzy criteria given by a decision-maker. The proposed method can be applied especially in big-data environments, because it avoids a detailed evaluation of every alternative in a huge solution space. At the first stage, we

TABLE II
FULL DECISION TABLE (FRAGMENT)

	a_1			a_2					a_3		
	A_{11}	A_{12}	A_{13}	A_{21}	A_{22}	A_{23}	A_{24}	A_{25}	A_{31}	A_{32}	A_{33}
x_1	0.2	0.8	0.0	0.0	0.3	0.7	0.0	0.0	0.0	0.3	0.7
x_2	0.2	0.8	0.0	0.0	0.0	0.3	0.7	0.0	0.2	0.8	0.0
...
x_{15}	0.8	0.2	0.0	0.7	0.3	0.0	0.0	0.0	0.0	0.0	1.0

TABLE III
DEGREES OF PREFERENCE FOR THE LINGUISTIC VALUES OF CRITERIA

A_{11}	a_1		a_2					a_3		
	A_{12}	A_{13}	A_{21}	A_{22}	A_{23}	A_{24}	A_{25}	A_{31}	A_{32}	A_{33}
1.00	0.50	0.25	0.00	0.50	1.00	0.75	0.25	0.00	0.50	1.00

TABLE IV
WEIGHTED COMPATIBILITIES OF LINGUISTIC LABELS AND ALTERNATIVES

Position of label	Label	Weighted compatibility of label	Alternative	Weighted compatibility of alternative	Order of alternatives
1	L_2	1.00	x_8 x_{14}	0.833 0.908	x_{14}, x_8
2	L_1	0.83	x_1 x_4 x_7 x_{12}	0.767 0.742 0.792 0.717	x_7, x_1, x_4, x_{12}
3	L_5	0.75	x_9 x_{10}	0.692 0.708	x_{10}, x_9
4	L_7	0.67	x_{15}	0.683	x_{15}
5	L_3	0.58	x_2 x_5	0.558 0.550	x_2, x_5
6	L_4	0.50	x_3 x_6	0.500 0.517	x_6, x_3
7	L_6	0.33	x_{11} x_{13}	0.333 0.433	x_{13}, x_{11}

evaluate only the linguistic labels, which represent classes of similar alternatives. Only a small number of the promising similarity classes of alternatives are selected for a detailed evaluation of alternatives at the second stage. Implementation of the presented method is simple, and obtained results can be easily interpreted. In future work, we plan extend the proposed method to apply both subjective linguistic and objective numerical criteria.

REFERENCES

- [1] S. Greco, M. Ehrgott, and J. R. Figueira, *Multiple Criteria Decision Analysis: State of the Art Surveys*. New York: Springer-Verlag, 2016.
- [2] A. M. Radzikowska and E. E. Kerre, "A comparative study of fuzzy rough sets," *Fuzzy Sets and Systems*, vol. 126, pp. 137–155, 2002.
- [3] L. D'eer and C. Cornelis, "A comprehensive study of fuzzy covering-based rough set models: Definitions, properties and interrelationships," *Fuzzy Sets and Systems*, vol. 336, pp. 1–26, 2018.
- [4] F. Cabrerizo, W. Pedrycz, I. Perez, S. Alonso, and E. Herrera-Viedma, "Group decision making in linguistic contexts: An information granulation approach," *Procedia Computer Science*, vol. 91, pp. 715–724, 2016.
- [5] S.-J. Chuu, "Interactive group decision-making using a fuzzy linguistic approach for evaluating the flexibility in a supply chain," *European Journal of Operational Research*, vol. 213, no. 1, pp. 279–289, 2011.
- [6] W. Pedrycz, P. Ekel, and R. Parreiras, *Fuzzy Multicriteria Decision-Making: Models, Methods and Applications*. Chichester: John Wiley & Sons Ltd, 2011.
- [7] C. Kahraman, S. C. Onar, and B. Oztaysi, "Fuzzy multicriteria decision-making: A literature review," *International Journal of Computational Intelligence Systems*, vol. 8, no. 4, pp. 637–666, 2015.
- [8] A. Mieszkowicz-Rolka and L. Rolka, "Labeled fuzzy rough sets versus fuzzy flow graphs," in *Proceedings of the 8th International Joint Conference on Computational Intelligence – Volume 2: FCTA*, J. J. Merelo et al., Eds. SCITEPRESS Digital Library, 2016, pp. 115–120.
- [9] —, "A novel approach to fuzzy rough set-based analysis of information systems," in *Information Systems Architecture and Technology. Knowledge Based Approach to the Design, Control and Decision Support*, ser. Advances in Intelligent Systems and Computing, Z. Wilimowska et al., Eds., vol. 432. Switzerland: Springer International Publishing, 2016, pp. 173–183.
- [10] Z. Pawlak, *Rough Sets: Theoretical Aspects of Reasoning about Data*. Boston Dordrecht London: Kluwer Academic Publishers, 1991.

Fuzzy Quantifier-Based Fuzzy Rough Sets

Adnan Theerens*, Chris Cornelis*

*Ghent University, Ghent, Belgium

Computational Web Intelligence

Dept. of Applied Mathematics, Computer Science and Statistics

Email: {adnan.theerens, chris.cornelis}@ugent.be

Abstract—In this paper we apply vague quantification to fuzzy rough sets to introduce *fuzzy quantifier-based fuzzy rough sets (FQFRS)*, an intuitive generalization of fuzzy rough sets. We show how several existing models fit in this generalization as well as how it inspires novel models that may improve these existing models. In addition, we introduce several new binary quantification models. Finally, we introduce an adaptation of FQFRS that allows seamless integration of outlier detection algorithms to enhance the robustness of the applications based on FQFRS.

I. INTRODUCTION

FUZZY quantification is an important part of fuzzy logic that models quantified sentences such as “Most Dutch people are tall” and “Nearly half of the S&P 500 stocks are down 10%”. Quantifiers are an effective tool to describe the quantity of elements that satisfy a certain condition. This is especially true if the condition is of a vague nature, as for example in the quantified sentence “Most Dutch people are tall”, since the quantity of elements satisfying a fuzzy condition (being tall) is hard to assess. The two most studied types of quantifiers are unary and binary quantifiers, unary quantifiers being of the form “ Q_1 elements are A ” (e.g. “Some people are tall”) and binary quantifiers being of the form “ Q_2 A ’s are B ’s” (e.g. “Most Dutch people are tall”). The first evaluation method for fuzzy quantified statements was introduced by Zadeh [1]. His idea was to define a cardinality measure for fuzzy sets to evaluate the quantity of elements satisfying a condition. The problem with this approach is that the cardinality measure is cumulative, implying that a situation involving two people with a degree of tallness of 0.5 is regarded the same as one with one tall person (tallness 1) and one short person (tallness 0). An improved evaluation method was proposed by Yager [2], which is based on the Ordered Weighted Averaging (OWA) operator. This method is semantically more reasonable for unary quantifiers but still lacks soundness for binary quantifiers. To resolve these issues, Glöckner [3] developed a general framework for fuzzy quantification. In this framework, fuzzy quantifiers are fully determined by how they act on classical (i.e. non-fuzzy) sets and by the choice of a quantifier fuzzification mechanism (QFM). A QFM thus reduces the evaluation of any quantified statement to the evaluation of quantified statements with crisp arguments.

Rough set theory, introduced by Pawlak [4], provides a lower and upper approximation of a concept with respect to

the indiscernibility relation between objects. The lower and upper approximation contain all objects that are certainly, respectively possibly part of the concept. That is to say, an element is a member of the lower approximation of a concept if every element indiscernible from it belongs to the concept; and an element is a member of the upper approximation of the concept if there exists an element indiscernible from it that belongs to the concept. Rough set theory was first extended to fuzzy rough set theory by Dubois and Prade [5], where both the concept and the indiscernibility relation can be fuzzy. Fuzzy rough set theory has been used successfully for classification and other machine learning purposes, such as feature and instance selection [6], but due to the fact that the approximations in classical fuzzy rough sets are determined using the minimum and maximum operators, these approximations (and the applications based on them) are sensitive to noisy and outlying samples. To mitigate this problem, many noise-tolerant versions of fuzzy rough sets (FRS) have been proposed, such as Vaguely Quantified FRS (VQFRS) [7], β -Precision FRS [8], [9], Variable Precision FRS [10], Variable Precision (θ, σ)-FRS [11], Soft Fuzzy Rough Sets [12], Automatic Noisy Sample Detection FRS [13], Data-Distribution-Aware FRS [14], Probability Granular Distance based FRS [15], Ordered Weighted Averaging (OWA) based FRS (OWAFRS) [16] and Choquet-based FRS (CFRS) [17]. VQFRS and, as noted in [17], OWAFRS and CFRS are fuzzy rough set models based on vague quantification. In this paper, we introduce a generalization of fuzzy rough sets, called fuzzy quantifier-based fuzzy rough sets (FQFRS), that takes the idea behind VQFRS and CFRS one step further. It does this by using binary and unary quantification models to determine the lower and upper approximation of a concept, respectively. Furthermore, we explain how to adapt FQFRS to use normalized outlier scores [18] to boost the robustness of the lower and upper approximations in fuzzy rough sets.

This paper is structured as follows: in Section II, we recall the required prerequisites for (Choquet-based) fuzzy rough sets and vague quantification. Section III discusses different binary quantification models and introduces several new ones. In Section IV, fuzzy quantifier-based fuzzy rough sets (FQFRS) and confidence-based FQFRS are introduced and their relation with existing models is discussed as well as the possible benefits they may have. Sections V and VI conclude this paper and describe opportunities for future research.

II. PRELIMINARIES

A. Fuzzy logic

In this subsection, we recall the necessary notions of fuzzy set and fuzzy logical connectives. We start with the definition of a fuzzy set and a fuzzy relation.

Definition II.1. [19] A fuzzy set or membership function A on X is a function from X to the unit interval, i.e. $A : X \rightarrow [0, 1]$. The value $A(x)$ of an element $x \in X$ is called the degree of membership of x in the fuzzy set A . The set of all fuzzy sets on X is denoted as $\tilde{\mathcal{P}}(X)$.

Definition II.2. A fuzzy relation R on X is an element of $\tilde{\mathcal{P}}(X \times X)$. A fuzzy relation R is called reflexive if $R(x, x) = 1$ for every $x \in X$. For an element $y \in X$ and a fuzzy relation $R \in \tilde{\mathcal{P}}(X \times X)$, we define the R -foreset of y as the fuzzy set $Ry(x) := R(x, y)$.

We will also make use of conjunctors, implicators and negators which extend their Boolean counterparts to the fuzzy setting.

Definition II.3.

- A function $\mathcal{C} : [0, 1]^2 \rightarrow [0, 1]$ is called a conjunctor if it is increasing in both arguments and satisfies $\mathcal{C}(0, 0) = \mathcal{C}(0, 1) = 0$ and $\mathcal{C}(1, x) = x$ for all $x \in [0, 1]$. A commutative and associative conjunctor \mathcal{T} is called a t-norm. We will use the following notation $x \wedge_{\mathcal{C}} y := \mathcal{C}(x, y)$.
- A function $\mathcal{S} : [0, 1]^2 \rightarrow [0, 1]$ is called a t-conorm if it is non-decreasing in both arguments, commutative, associative, and satisfies $\mathcal{S}(0, x) = x$ for all $x \in [0, 1]$. We will use the following notation $x \vee_{\mathcal{S}} y := \mathcal{S}(x, y)$.
- A function $\mathcal{I} : [0, 1]^2 \rightarrow [0, 1]$ is called an impicator if $\mathcal{I}(0, 0) = \mathcal{I}(0, 1) = \mathcal{I}(1, 1) = 1$, $\mathcal{I}(1, 0) = 0$ and for all x_1, x_2, y_1, y_2 in $[0, 1]$ the following holds:

- 1) $x_1 \leq x_2 \Rightarrow \mathcal{I}(x_1, y_1) \geq \mathcal{I}(x_2, y_1)$ (non-increasing in the first argument),
- 2) $y_1 \leq y_2 \Rightarrow \mathcal{I}(x_1, y_1) \leq \mathcal{I}(x_1, y_2)$ (non-decreasing in the second argument),

We will use the following notation $x \rightarrow_{\mathcal{I}} y := \mathcal{I}(x, y)$.

- A function $\mathcal{N} : [0, 1] \rightarrow [0, 1]$ is called a negator if it is non-increasing and satisfies $\mathcal{N}(0) = 1$ and $\mathcal{N}(1) = 0$. A negator is called a strong negator if it is an involution.
- Suppose \mathcal{S} is a t-conorm and \mathcal{N} is a negator. The mapping

$$\mathcal{I}(x, y) = \mathcal{N}(x) \vee_{\mathcal{S}} y, \quad \forall x, y \in [0, 1],$$

is called the \mathcal{S} -implicator induced by \mathcal{S} and \mathcal{N} .

Example II.1. The Kleene-Dienes impicator is defined as $\mathcal{I}_{KD}(x, y) := \max(1 - x, y)$. It is the \mathcal{S} -implicator induced by the standard negator $\neg(x) := 1 - x$ and the standard t-conorm $x \vee y = \max(x, y)$.

Since t-norms are required to be associative, they can be extended naturally to a function $[0, 1]^n \rightarrow [0, 1]$ for any natural number $n \geq 2$.

Definition II.4. The notation $A \subseteq B$ for two fuzzy sets A and B , expresses that $A(x) \leq B(x)$ for all $x \in X$. The fuzzy set $A \cap B \in \tilde{\mathcal{P}}(X)$ is defined by $(A \cap B)(x) = \min(A(x), B(x))$. We denote Zadeh's Sigma count as $|A| := \sum_{x \in X} A(x)$ for every fuzzy set $A \in \tilde{\mathcal{P}}(X)$, it is a conservative extension of classical set cardinality to fuzzy sets.

Definition II.5. Given a negator \mathcal{N} , conjunctor \mathcal{C} , t-conorm \mathcal{S} , impicator \mathcal{I} , and two fuzzy sets $A, B \in \tilde{\mathcal{P}}(X)$, we define the following:

$$(\neg_{\mathcal{N}} A)(x) = \mathcal{N}(A(x)),$$

$$(A \cap_{\mathcal{C}} B)(x) := A(x) \wedge_{\mathcal{C}} B(x),$$

$$(A \cup_{\mathcal{S}} B)(x) := A(x) \vee_{\mathcal{S}} B(x),$$

$$(A \rightarrow_{\mathcal{I}} B)(x) := A(x) \rightarrow_{\mathcal{I}} B(x),$$

for all $x \in X$.

B. OWA-based fuzzy rough sets

A downside to the classical definition of lower and upper approximation in fuzzy rough set theory is their lack of robustness. The value of the membership of an element in the lower and upper approximation is fully determined by a single element because of the minimum and maximum operators in the definition. To solve this undesirable behaviour, many alternative definitions of fuzzy rough sets were introduced. One of these is OWA-based fuzzy rough sets [16], which has been shown to have an excellent trade-off between performance (robustness) and theoretical properties [20]. The Ordered Weighted Average [21] is an aggregation operator that is defined as follows:

Definition II.6 (OWA). Let $X = \{x_1, x_2, \dots, x_n\}$, $f : X \rightarrow \mathbb{R}$ and $\mathbf{w} = (w_1, w_2, \dots, w_n)$ be a weighting vector, i.e. $\mathbf{w} \in [0, 1]^n$ and $\sum_{i=1}^n w_i = 1$, then the ordered weighted average of f with respect to \mathbf{w} is defined as

$$OWA_{\mathbf{w}}(f) := \sum_{i=1}^n f(x_{\sigma(i)})w_i,$$

where σ is a permutation of $\{1, 2, \dots, n\}$ such that

$$f(x_{\sigma(1)}) \geq f(x_{\sigma(2)}) \geq \dots \geq f(x_{\sigma(n)}).$$

Example II.2. The maximum, mean and minimum operators can all be seen as OWA-operators with weight vectors $(1, 0, \dots, 0, 0)$, $(\frac{1}{n}, \frac{1}{n}, \dots, \frac{1}{n})$ and $(0, 0, \dots, 0, 1)$ respectively.

In OWA-based fuzzy rough sets, OWA operators replace the minimum and maximum in the lower and upper approximations in classical fuzzy rough sets. To not deviate too strongly from the original definitions, some requirements may be enforced on the weight vectors of the OWA-operators used [16]. In particular, the authors required that the OWA-operator for the lower approximation is a soft minimum and for the upper approximation a soft maximum.

Definition II.7. The orness and andness of a weight vector $\mathbf{w} = (w_i)_{i=1}^n$ are defined as

$$\text{orness}(\mathbf{w}) = \frac{1}{n-1} \sum_{i=1}^n ((n-i) \cdot w_i), \quad (1)$$

$$\text{andness}(\mathbf{w}) = 1 - \text{orness}(\mathbf{w}).$$

If $\text{orness}(\mathbf{w}) < 0.5$, then $OWA_{\mathbf{w}}$ is called a soft minimum. If $\text{orness}(\mathbf{w}) > 0.5$, $OWA_{\mathbf{w}}$ is called a soft maximum.

As can be seen from Equation (1), the orness indicates how much weight is given to the largest elements. The orness tells us how “close” the OWA-operator is to the maximum. Using this definition OWA-based fuzzy rough sets are then defined as:

Definition II.8. [16] Given $R \in \tilde{\mathcal{P}}(X \times X)$, weight vectors \mathbf{w}_l and \mathbf{w}_u with $\text{orness}(\mathbf{w}_l) < 0.5$ and $\text{orness}(\mathbf{w}_u) > 0.5$ and $A \in \tilde{\mathcal{P}}(X)$, the OWA lower and upper approximation of A w.r.t. R , \mathbf{w}_l and \mathbf{w}_u are given by:

$$(\underline{apr}_{R, \mathbf{w}_l} A)(x) = OWA_{\mathbf{w}_l}(\mathcal{I}(R(x, y), A(y))), \quad (2)$$

$$(\overline{apr}_{R, \mathbf{w}_u} A)(x) = OWA_{\mathbf{w}_u}(\mathcal{C}(R(x, y), A(y))), \quad (3)$$

where \mathcal{I} is an implicator, \mathcal{C} a conjunctor and $\mathcal{I}(R(x, y), A(y))$ and $\mathcal{C}(R(x, y), A(y))$ are seen as functions in y .

C. The Choquet integral

The Choquet integral induces a large class of aggregation functions, namely the class of all comonotone linear aggregation functions [22]. Since we will view the Choquet integral as an aggregation operator, we will restrict ourselves to measures (and Choquet integrals) on finite sets. For the general setting, we refer the reader to e.g. [23].

Definition II.9. Let X be a finite set. A function $\mu : \mathcal{P}(X) \rightarrow [0, 1]$ is called a monotone measure if:

- $\mu(\emptyset) = 0$ and $\mu(X) = 1$,
- $(\forall A, B \in \mathcal{P}(X))(A \subseteq B \implies \mu(A) \leq \mu(B))$.

A monotone measure is called:

- additive if $\mu(A \cup B) = \mu(A) + \mu(B)$ when A and B are disjoint,
- symmetric if $\mu(A) = \mu(B)$ when $|A| = |B|$.

Definition II.10. [23] Let μ be a monotone measure on X and $f : X \rightarrow \mathbb{R}$ a real-valued function. The Choquet integral of f with respect to the measure μ is defined as:

$$\int f d\mu = \sum_{i=1}^n \mu(A_i^*) \cdot [f(x_i^*) - f(x_{i-1}^*)],$$

where $(x_1^*, x_2^*, \dots, x_n^*)$ is a permutation of $X = \{x_1, x_2, \dots, x_n\}$ such that

$$f(x_1^*) \leq f(x_2^*) \leq \dots \leq f(x_n^*),$$

$A_i^* := \{x_i^*, \dots, x_n^*\}$ and $f(x_0^*) := 0$.

The class of aggregation operators induced by the Choquet integral contains the weighted mean and the OWA operator. In

fact, the weighted mean and OWA operator are the Choquet integrals with respect to additive and symmetric measures, respectively.

Proposition II.11. [22] The Choquet integral with respect to an additive measure μ is the weighted mean $M_{\mathbf{w}}$ with weight vector $\mathbf{w} = (w_i)_{i=1}^n = (\mu(\{x_i\}))_{i=1}^n$. Conversely, the weighted mean $M_{\mathbf{v}}$, with weight vector $\mathbf{v} = (v_i)_{i=1}^n$ is a Choquet integral with respect to the uniquely defined additive measure μ for which $(\mu(\{x_i\}))_{i=1}^n = (v_i)_{i=1}^n$.

Proposition II.12. [22] The Choquet integral with respect to a symmetric measure μ is the OWA operator with weight vector $\mathbf{w} = (w_i)_{i=1}^n = (\mu(A_i) - \mu(A_{i-1}))_{i=1}^n$, where A_i denotes any subset with cardinality i . Conversely, the OWA operator with weight vector $\mathbf{v} = (v_i)_{i=1}^n$ is a Choquet integral with respect to the symmetric measure μ defined as

$$(\forall A \subseteq X)(\mu(A)) := \sum_{i=1}^{|A|} v_i.$$

D. Glöckner’s framework for fuzzy quantification

Glöckner’s framework for fuzzy quantification deals with defining vague quantifiers in two steps. The first step is the specification of the vague quantifier on crisp sets, i.e. to specify the “underlying” semi-fuzzy quantifier. The second step is to extend this description to fuzzy arguments, i.e. applying a quantifier fuzzification mechanism.

Definition II.13. [3] An n -ary semi-fuzzy quantifier on $X \neq \emptyset$ is a mapping $Q : (\mathcal{P}(X))^n \rightarrow [0, 1]$. An n -ary fuzzy quantifier on $X \neq \emptyset$ is a mapping $\tilde{Q} : (\tilde{\mathcal{P}}(X))^n \rightarrow [0, 1]$.

Definition II.14. [3] A quantifier fuzzification mechanism (QFM) \mathcal{F} assigns to each semi-fuzzy quantifier $Q : (\mathcal{P}(X))^n \rightarrow [0, 1]$ a corresponding fuzzy quantifier $\mathcal{F}(Q) : (\mathcal{P}(X))^n \rightarrow [0, 1]$ of the same arity $n \in \mathbb{N}$ and on the same universe X .

Glöckner defined an axiomatic framework for plausible models of fuzzy quantification which he called the Determiner Fuzzification Scheme (DFS) axioms. Since introducing DFS would take up too much space we refer the reader to chapter three, four, and five of [3].

We now take a look at Zadeh’s and Yager’s traditional approaches, where they describe fuzzy quantifiers using fuzzy sets of the unit interval.

Definition II.15. [1] A fuzzy set $\Lambda \in \tilde{\mathcal{P}}([0, 1])$ is called a regular increasing monotone (RIM) quantifier if Λ is a non-decreasing function such that $\Lambda(0) = 0$ and $\Lambda(1) = 1$.

Example II.3. The following RIM quantifiers represent the quantifiers “more than $100 * k\%$ ” and “at least $100 * k\%$ ”:

$$\Lambda_{>k}(p) = \begin{cases} 1 & \text{if } p > k \\ 0 & \text{elsewhere} \end{cases} \quad \Lambda_{\geq k}(p) = \begin{cases} 1 & \text{if } p \geq k \\ 0 & \text{elsewhere} \end{cases}$$

These RIM quantifiers also include (a representation of) the universal and existential quantifier, $\Lambda_{\forall} := \Lambda_{>0}$ and $\Lambda_{\exists} :=$

$\Lambda_{\geq 1}$. Linguistic quantifiers such as “most” and “some” can be modelled using Zadeh’s S-function ($0 \leq \alpha < \beta \leq 1$):

$$\Lambda_{(\alpha, \beta)}(p) = \begin{cases} 0 & p \leq \alpha \\ \frac{2(p-\alpha)^2}{(\beta-\alpha)^2} & \alpha \leq p \leq \frac{\alpha+\beta}{2} \\ 1 - \frac{2(p-\beta)^2}{(\beta-\alpha)^2} & \frac{\alpha+\beta}{2} \leq p \leq \beta \\ 1 & \beta \leq p \end{cases},$$

for example, we could use $\Lambda_{(0.3, 0.9)}$ and $\Lambda_{(0.1, 0.4)}$ to model “most” and “some”, respectively.

In Zadeh’s model, unary sentences of the form “ Λ X ’s are A ’s” and binary sentences of the form “ Λ A ’s are B ’s”, where Λ is a RIM quantifier and $A, B \in \tilde{\mathcal{P}}(X)$, are evaluated as

$$\tilde{Z}_{\Lambda}(A) = \Lambda\left(\frac{|A|}{|X|}\right), \quad (4)$$

$$\tilde{Z}_{\Lambda}^2(A, B) = \Lambda\left(\frac{|A \cap B|}{|A|}\right), \quad (5)$$

respectively, while in Yager’s OWA model, the unary sentence “ Λ X ’s are A ’s” is evaluated as

$$\tilde{Y}_{\Lambda}(A) := OWA_{\mathbf{w}^{\Lambda}}(A), \quad (6)$$

where

$$w_i^{\Lambda} := \Lambda\left(\frac{i}{n}\right) - \Lambda\left(\frac{i-1}{n}\right). \quad (7)$$

For the binary sentence “ Λ A ’s are B ’s”, there is no definite evaluation, although there are two that are most common in literature (cf. [24], [25]). The first one evaluates the sentence as

$$\tilde{Y}_{\Lambda}^{\mathcal{I}}(A, B) := \tilde{Y}_{\Lambda}(\mathcal{I}(A, B)) = OWA_{\mathbf{w}^{\Lambda}}(\mathcal{I}(A, B)), \quad (8)$$

where \mathcal{I} is an implicator, while the second one evaluates it as:

$$\tilde{Y}_{\Lambda}^2(A, B) := OWA_{\mathbf{v}^{\Lambda}}(\mathcal{I}(A, B)), \quad (9)$$

where

$$v_i := \Lambda\left(\frac{\sum_{j=1}^i A(x_j^*)}{|A|}\right) - \Lambda\left(\frac{\sum_{j=1}^{i-1} A(x_j^*)}{|A|}\right), \quad (10)$$

with $A(x_i^*)$ being the i th smallest $A(x)$ for $x \in X$ and $\sum_{j=1}^0 A(x_j^*) = \sum_{x \in \emptyset} x = 0$. Note that both \tilde{Z}_{Λ} and \tilde{Y}_{Λ} extend the semi-fuzzy quantifier

$$Q_{\Lambda}(A) := \Lambda\left(\frac{|A|}{|X|}\right). \quad (11)$$

E. Choquet-based fuzzy rough sets

Choquet-based fuzzy rough sets (CFRS) [17] have been introduced by noting that by Proposition II.12, we can rewrite OWAFRS as follows:

$$(\underline{\text{apr}}_{R, \mu_l} A)(y) = \int \mathcal{I}(R(x, y), A(x)) d\mu_l(x),$$

$$(\overline{\text{apr}}_{R, \mu_u} A)(y) = \int \mathcal{C}(R(x, y), A(x)) d\mu_u(x),$$

where μ_l and μ_u are two symmetric measures. Allowing non-symmetric measures gives us the definition of CFRS:

Definition II.16. [17] Given $R \in \tilde{\mathcal{P}}(X \times X)$, monotone measures μ_l and μ_u on X and $A \in \tilde{\mathcal{P}}(X)$, then the Choquet lower and upper approximation of A w.r.t. R , μ_l and μ_u are given by:

$$(\underline{\text{apr}}_{R, \mu_l} A)(y) = \int \mathcal{I}(R(x, y), A(x)) d\mu_l(x) \quad (12)$$

$$(\overline{\text{apr}}_{R, \mu_u} A)(y) = \int \mathcal{C}(R(x, y), A(x)) d\mu_u(x), \quad (13)$$

where \mathcal{I} is an implicator and \mathcal{C} is a conjunctor.

Example II.4. Suppose we have a crisp set O containing all the instances that are outliers, unreliable or inaccurate, then a useful pair of quantifiers could be “for all except (maybe) elements of O ” and “there exists an element in $X \setminus O$ ”. These quantifiers can be modelled by the partial minimum and maximum, which in turn are Choquet-integral operators with respect to non-symmetric measures (cf. [17]).

Using these non-symmetric measures in Equation (12) and (13), we get that the degree of membership of an element y to the lower approximation is equal to the truth value of the proposition “All trustworthy elements that are indiscernible to y are in A ”. An analogous interpretation holds for the upper approximation.

As we will see in Subsection II-F, it is possible to extend this approach of the previous example to fuzzy sets O and quantifiers representing “most of the trustworthy objects”. The following examples show how such fuzzy sets O can be constructed in practice.

Example II.5. Suppose we have a decision system $(X, A \cup \{d\})$ where d is a categorical attribute. Then we can define $O(x)$ as the normalized outlier score [18] of x (obtained from a certain outlier detection algorithm) when compared to other elements of $[x]_d$ (based on the conditional attributes). An outlier score measures the degree to which a data point differs from other observations, and normalization transforms this score in such a way that it can be interpreted as a degree of outlieriness.

Example II.6. Suppose X consists of patients from several different hospitals, A is the subset of patients that have a disease and R is a similarity relation between patients based on a set of symptoms. Then a confidence score c_i can be attached to each hospital i based on the accuracy of the tests performed to trace the disease (and the symptoms). The membership degree of a patient x of hospital i to O can then be defined as $O(x) = 1 - c_i$.

F. Examples of non-symmetric measures

As described in the previous subsection we can accommodate non-symmetry by introducing a fuzzy set O in X that represents the lack of confidence. The value $O(x)$ could, for example, be seen as an outlier score in $[0, 1]$ (Example II.5) or it could represent the unreliability or inaccuracy of the observation (Example II.6). We now recall several non-symmetric measures that were introduced in [17] using the fuzzy set O .

1) *Fuzzy removal*: the first option that was proposed to define a non-symmetric measure using O is as follows:

$$\mu_{\forall x \in X \setminus O}(S) = \begin{cases} 1 & \text{if } S = X \\ 0 & \text{if } S = \emptyset \\ \mathcal{T}(\underbrace{O(x)}_{x \in X \setminus S}) & \text{elsewhere} \end{cases}, \quad (14)$$

where \mathcal{T} is a t-norm (e.g. minimum) and $S \in \mathcal{P}(X)$. This measure is called the *fuzzy removal* measure, since in the case O is crisp, the Choquet integral with respect to this measure is equal to the partial minimum. The vague quantifier interpretation of the fuzzy removal measure could thus be “for all except (maybe) elements of O ”.

2) *Weighted ordered weighted average*: the second idea for a non-symmetric measure was:

$$\mu_{\Lambda}^{-O}(S) := \Lambda \left(\frac{|\neg O \cap S|}{|\neg O|} \right) = \Lambda \left(\sum_{x_i \in S} p_i \right), \quad (15)$$

where Λ is a RIM quantifier and \mathbf{p} a weight vector describing the confidence, reliability, accuracy or non-outlierness of each observation:

$$p_i = \frac{1 - O(x_i)}{n - \sum_{j=1}^n O(x_j)}. \quad (16)$$

The measure in Equation (15) corresponds to the Weighted Ordered Weighted Averaging (WOWA) operator [26], [27], which is a generalization of the OWA and the weighted mean. The RIM quantifier Λ determines the OWA part of the WOWA and the weight vector \mathbf{p} the weighted mean part. The WOWA operator is also equivalent with Yager’s importance weighted quantifier guided aggregation [2]. These measures could be interpreted as quantifiers of the form “ Λ of the trustworthy/reliable objects”.

III. BINARY QUANTIFICATION MODELS

We now take a deeper look at binary quantifiers (i.e. 2-ary fuzzy quantifiers), and in particular, proportional quantifiers such as “Most A ’s are B ’s”. To focus our attention we will make use of Zadeh’s approach of using RIM-quantifiers to model these quantifiers in their bare form.

A. QFM-based binary quantification models

To define binary fuzzy quantifiers using QFM’s, we first need semi-fuzzy quantifiers. The following definition proposes the two most viable options for semi-fuzzy quantifiers that model “ Λ A ’s are B ’s”, with Λ a RIM-quantifier.

Definition III.1. *Given a RIM-quantifier Λ , we define the following semi-fuzzy quantifiers:*

$$Q_{\Lambda}^2(A, B) := \Lambda \left(\frac{|A \cap B|}{|A|} \right),$$

$$Q_{\Lambda}^{\rightarrow}(A, B) = \Lambda \left(\frac{|A \rightarrow B|}{|X|} \right) := \Lambda \left(\frac{|\neg A| + |A \cap B|}{|X|} \right),$$

for crisp sets $A, B \in \mathcal{P}(X)$.

The first one is the most intuitive definition, but the second one is (as we will see) used a lot in practice, perhaps due to

its simplicity as we shall see later in this section (Corollary III.6). The semantical difference between Q_{Λ}^2 and $Q_{\Lambda}^{\rightarrow}$ is that of “Most A ’s are B ’s” and of “For most X ’s, if they are in A , they are in B ”. This is a very subtle difference and in day to day life both mean the same. In the first one only elements of A matter, while for the second one all elements matter. The following example demonstrates how important the difference is.

Example III.1. *Let us look at the difference between “Most Belgian people are not Belgian” and “For most people in the world, if they are Belgian, they are not Belgian”. Most people would say both sentences are plainly wrong. Let X denote the set of all people and $B \in \mathcal{P}(X)$ the subset of Belgian people, if we evaluate the first sentence using Q_{Λ} and the second one using $Q_{\Lambda}^{\rightarrow}$, we get the following:*

$$Q_{\Lambda}^2(B, \neg B) = \Lambda \left(\frac{|\emptyset|}{|B|} \right) = 0,$$

$$Q_{\Lambda}^{\rightarrow}(B, \neg B) = \Lambda \left(\frac{|\neg B|}{|X|} \right) \approx 1,$$

because the percentage of Belgians in the world is minuscule. So the second one is still correct, since for most people it holds that if they are Belgian, then they are not Belgian, since they are simply not from Belgium.

If \mathcal{F} is a QFM, we can use $\mathcal{F}(Q_{\Lambda}^2)$ and $\mathcal{F}(Q_{\Lambda}^{\rightarrow})$ to evaluate sentences of the form “ Λ A ’s are B ’s” for $A, B \in \tilde{\mathcal{P}}(X)$. We now take a look at the differences between the two. The first difference is the monotonicity. In the second argument both Q_{Λ} and $Q_{\Lambda}^{\rightarrow}$ are non-decreasing, hence so are $\mathcal{F}(Q_{\Lambda})$ and $\mathcal{F}(Q_{\Lambda}^{\rightarrow})$ for a DFS \mathcal{F} (argument monotonicity [3]). The difference between the two is in the first argument; let us add an element a to A and suppose Λ is a strictly increasing RIM-quantifier, if $a \in B$, then $Q_{\Lambda}^2(A, B)$ will strictly increase, while $Q_{\Lambda}^{\rightarrow}(A, B)$ stays unchanged, if $a \notin B$, then $Q_{\Lambda}^2(A, B)$ and $Q_{\Lambda}^{\rightarrow}(A, B)$ will strictly decrease. So in summary, $Q_{\Lambda}^{\rightarrow}$ is non-increasing in the first argument (hence $\mathcal{F}(Q_{\Lambda}^{\rightarrow})$ is as well), while Q_{Λ}^2 is not monotone in the first argument.

Proposition III.2. *We have the following inequality:*

$$Q_{\Lambda}^{\rightarrow}(A, B) \geq Q_{\Lambda}^2(A, B),$$

for every $A, B \in \mathcal{P}(X)$.

Proof.

$$\begin{aligned} & \Lambda \left(\frac{|\neg A| + |A \cap B|}{|X|} \right) \geq \Lambda \left(\frac{|A \cap B|}{|A|} \right) \\ \iff & \frac{|\neg A| + |A \cap B|}{|\neg A| + |A|} \geq \frac{|A \cap B|}{|A|} \\ \iff & (|\neg A| + |A \cap B|) * |A| \geq |A \cap B| * (|\neg A| + |A|) \\ \iff & |\neg A| * |A| \geq |A \cap B| * |\neg A| \\ \iff & |A| \geq |A \cap B| \end{aligned}$$

□

Corollary III.3. We have the following inequality for every DFS \mathcal{F} :

$$\mathcal{F}(Q_{\Lambda}^{\rightarrow})(A, B) \geq \mathcal{F}(Q_{\Lambda}^2)(A, B),$$

for every $A, B \in \tilde{\mathcal{P}}(X)$.

Proof. Every DFS satisfies quantifier monotonicity [3]. \square

We will now show that evaluating the binary quantifier $\mathcal{F}(Q_{\Lambda}^{\rightarrow})$ for fuzzy sets A, B and a DFS \mathcal{F} simply amounts to evaluating the fuzzy set $A \rightarrow B$ using the unary quantifier $\mathcal{F}(Q_{\Lambda})$, where \rightarrow is the implicator induced by the DFS \mathcal{F} (cf. [3]).

Definition III.4. Let $\tilde{Q} : (\tilde{\mathcal{P}}(X))^n \rightarrow [0, 1]$ be a fuzzy quantifier, then the fuzzy quantifier $Q \rightarrow : (\tilde{\mathcal{P}}(X))^{n+1} \rightarrow [0, 1]$ is defined as:

$$\tilde{Q} \rightarrow (A_1, \dots, A_{n+1}) := \tilde{Q}(A_1, \dots, A_{n-1}, (A_n \rightarrow A_{n+1})).$$

For a semi-fuzzy quantifier Q the semi-fuzzy quantifier $Q \rightarrow$ is defined analogously.

Proposition III.5. For every semi-fuzzy quantifier Q and DFS \mathcal{F} we have:

$$\mathcal{F}(Q \rightarrow) = \mathcal{F}(Q) \rightarrow.$$

Proof. This follows from

$$Q \rightarrow = Q \cap \neg$$

and the fact that a DFS is compatible with internal meets and internal negations [3]. \square

Corollary III.6. Let \mathcal{F} be a DFS and Q_{Λ} the unary quantifier from Equation (11), then:

$$\mathcal{F}(Q_{\Lambda}^{\rightarrow})(A, B) = \mathcal{F}(Q_{\Lambda})(A \rightarrow B),$$

for every $A, B \in \tilde{\mathcal{P}}(X)$.

Applying this to one of the most used QFM's, Glöckner's \mathcal{F}_{owa} [3], we can write one of Yager's binary quantification models as a QFM-based model:

Corollary III.7.

$$\mathcal{F}_{owa}(Q_{\Lambda}^{\rightarrow})(A, B) = \int \mathcal{I}_{KD}(A, B) d\mu_{\Lambda} = \tilde{Y}_{\Lambda}^{\mathcal{I}_{KD}}(A, B)$$

Proof. Follows from the fact that \mathcal{F}_{owa} is a standard DFS (thus the induced implicator is \mathcal{I}_{KD}) [3] and

$$\mathcal{F}_{owa}(Q)(A) = \int A dQ,$$

for every $A \in \tilde{\mathcal{P}}(X)$ and every semi-fuzzy quantifier Q that is also a monotone measure [3]. \square

Remark III.2. We can apply the exact same reasoning as in the previous corollary to the standard DFS \mathcal{M}_{CX} [3], to obtain

$$\mathcal{M}_{CX}(Q_{\Lambda}^{\rightarrow})(A, B) = (S) \int \mathcal{I}_{KD}(A, B) d\mu_{\Lambda},$$

where $(S) \int$ denotes the Sugeno integral [23].

B. Integral-based binary quantification models

We start off with rewriting Zadeh's and Yager's models using the Choquet integral to get a unifying view of these models.

Proposition III.8. Let Λ be a RIM-quantifier, $A, B \in \tilde{\mathcal{P}}(X)$ and \mathcal{I} an implicator. We can rewrite Zadeh's and Yager's evaluation models as follows:

$$\tilde{Z}_{\Lambda}(A) = \Lambda \left(\int A d\mu_{id} \right)$$

$$\tilde{Z}_{\Lambda}^2(A, B) = \Lambda \left(\frac{\int A \cap B d\mu_{id}}{\int A d\mu_{id}} \right)$$

$$\tilde{Y}_{\Lambda}(A) = \int A d\mu_{\Lambda}$$

$$\tilde{Y}_{\Lambda}^{\mathcal{I}}(A, B) = \int \mathcal{I}(A, B) d\mu_{\Lambda}$$

$$\tilde{Y}'_{\Lambda}(A, B) = \int \mathcal{I}(A, B) d\mu'_{\Lambda},$$

where

$$\mu_{\Lambda}(S) := \Lambda \left(\frac{|S|}{|X|} \right)$$

$$\mu'_{\Lambda}(S) := \Lambda \left(\frac{\sum_{j=1}^{|S|} A(x_j^*)}{|A|} \right),$$

with $A(x_i^*)$ being the i th smallest $A(x)$ for $x \in X$ and $S \in \mathcal{P}(X)$.

Proof. Follows from Propositions II.11 and II.12. \square

So both models can be written using integrals, but whereas Zadeh's model first integrates and then applies the RIM quantifier, Yager's model already incorporates the RIM quantifier in the measure used for integration.

Looking at Yager's model $\tilde{Y}_{\Lambda}^{\mathcal{I}}$ we can see that an element not in A contributes as much to the truth value as an element that is in A and in B , therefore the model has the same issues as mentioned in Example III.1. Quantifiers based on the semi-fuzzy quantifier Q_{Λ}^2 and a DFS do not suffer from this issue, but are computationally more complex. Therefore we now introduce a new binary quantification model that does an extra weighting on elements of A to compensate for the issues of $\tilde{Y}_{\Lambda}^{\mathcal{I}}$:

Definition III.9. Let Λ be a RIM-quantifier, \mathcal{I} an implicator and $A, B \in \tilde{\mathcal{P}}(X)$. We define the fuzzy quantifier $\tilde{W}_{\Lambda}^{\mathcal{I}} : \tilde{\mathcal{P}}(X) \rightarrow [0, 1]$ as:

$$\tilde{W}_{\Lambda}^{\mathcal{I}}(A, B) := \int \mathcal{I}(A, B) d\mu_{\Lambda}^A, \quad \mu_{\Lambda}^A(S) := \Lambda \left(\frac{|S \cap A|}{|A|} \right),$$

where $S \in \mathcal{P}(X)$.

Remark III.3. By replacing the Choquet integral in the previous definition with the Sugeno integral, or even general pan-integrals (for more information about these integrals see [23]), we obtain other novel quantifiers that are worth studying. Because Glöckner's model \mathcal{M}_{CX} corresponds to the Sugeno

integral (Remark III.2), this can give us a good compromise (between computational simplicity and semantical soundness) for the $\mathcal{M}_{CX}(Q_\Lambda^2)$ model.

This quantifier indeed resembles $\tilde{Y}_\Lambda^{\mathcal{I}}(A, B)$ but with an extra weighting on A . The quantifier \tilde{Y}_Λ^2 also does this but in a less intuitive and elegant way, by doing an extra ordering of the elements. Comparing the two, we see that the new definition uses a weighted ordered weighted averaging (WOWA) with Λ for the OWA part and $p_i = A(x_i)/|A|$ for the weighted mean part, while the quantifier \tilde{Y}_Λ^2 uses an OWA operator with weights given by Equation (10).

The following proposition describes how all of these fuzzy quantifiers act on crisp sets, i.e., what their underlying semi-fuzzy quantifiers are.

Proposition III.10. *The following equalities hold:*

$$\begin{aligned}\tilde{W}_\Lambda^{\mathcal{I}}(A, B) &= Q_\Lambda(A, B), \\ \tilde{Y}_\Lambda^2(A, B) &= Q_\Lambda(A, B), \\ \tilde{Z}_\Lambda^2(A, B) &= Q_\Lambda(A, B), \\ \tilde{Y}_\Lambda^{\mathcal{I}}(A, B) &= Q_\Lambda^{\rightarrow}(A, B),\end{aligned}$$

for all crisp sets $A, B \in \mathcal{P}(X)$.

Proof. We will only prove the first and second equality, the rest are analogous or trivial. Let $A, B \in \mathcal{P}(X)$ be two crisp sets, then:

$$\begin{aligned}\mathcal{U}(\tilde{W}_\Lambda^{\mathcal{I}})(A, B) &= \int \mathcal{I}(A, B) d\mu_\Lambda^A = \mu_\Lambda^A(\mathcal{I}(A, B)) \\ &= \mu_\Lambda^A(\neg A \cup B) \\ &= \Lambda\left(\frac{|A \cap B|}{|A|}\right) = Q_\Lambda(A, B),\end{aligned}$$

which proves the first equality. For the second equality:

$$\begin{aligned}\mathcal{U}(\tilde{Y}_\Lambda^2)(A, B) &= \int \mathcal{I}(A, B) d\mu'_\Lambda = \mu'_\Lambda(\mathcal{I}(A, B)) \\ &= \mu'_\Lambda(\neg A \cup B) \\ &= \mu'_\Lambda(\neg A \cup (A \cap B))\end{aligned}$$

But for crisp sets A the measure μ'_Λ reduces to:

$$\mu'_\Lambda(S) = \begin{cases} 0 & \text{if } |S| \leq |\neg A| \\ \frac{|S| - |\neg A|}{|A|} & \text{if } |S| > |\neg A| \end{cases},$$

from which we get the desired

$$\begin{aligned}\mathcal{U}(\tilde{Y}_\Lambda^2)(A, B) &= \mu'_\Lambda(\neg A \cup (A \cap B)) \\ &= \Lambda\left(\frac{|\neg A \cup (A \cap B)| - |\neg A|}{|A|}\right) \\ &= \Lambda\left(\frac{|A \cap B|}{|A|}\right).\end{aligned}$$

□

The previous proposition thus shows that both \tilde{Y}_Λ^2 and $\tilde{W}_\Lambda^{\mathcal{I}}$ apply the correct weighting on A such that for crisp arguments the quantifiers act intuitively.

IV. FUZZY QUANTIFIER-BASED FUZZY ROUGH SETS

Let us take another look at OWAFRS and rewrite its approximations as follows:

$$\begin{aligned}(\underline{\text{apr}}_{R, \mu_l} A)(y) &= \int \mathcal{I}(R(x, y), A(x)) d\mu_l(x), \\ &= \tilde{Y}_\Lambda^{\mathcal{I}}(Ry, A)\end{aligned}\quad (17)$$

$$\begin{aligned}(\overline{\text{apr}}_{R, \mu_u} A)(y) &= \int \mathcal{C}(R(x, y), A(x)) d\mu_u(x) \\ &= \tilde{Y}_\Upsilon(Ry \cap_C A),\end{aligned}\quad (18)$$

where μ_l and μ_u are symmetric measures, and Λ and Υ are their corresponding RIM-quantifiers (cf. Section 3.4 in [17]). Thus, the lower and upper approximations of OWAFRS are evaluated by evaluating vaguely quantified propositions using Yager's quantification model ($\tilde{Y}_\Lambda^{\rightarrow}$ and \tilde{Y}_Λ). We now introduce fuzzy quantifier-based fuzzy rough sets (FQFRS) by allowing general (binary for lower approximation and unary for upper approximation) quantification models.

Definition IV.1 ($(\tilde{Q}_l, \tilde{Q}_u)$ -fuzzy rough set). *Given a reflexive fuzzy relation $R \in \mathcal{F}(X \times X)$, fuzzy quantifiers $\tilde{Q}_l : (\tilde{\mathcal{P}}(X))^2 \rightarrow [0, 1]$ and $\tilde{Q}_u : \tilde{\mathcal{P}}(X) \rightarrow [0, 1]$, and $A \in \mathcal{F}(X)$, then the lower and upper approximation of A w.r.t. R are given by:*

$$\begin{aligned}(\underline{\text{apr}}_{R, \tilde{Q}_l} A)(y) &= \tilde{Q}_l(Ry, A), \\ (\overline{\text{apr}}_{R, \tilde{Q}_u} A)(y) &= \tilde{Q}_u(Ry \cap_C A),\end{aligned}$$

where \mathcal{C} is a conjunctor.

Suppose \tilde{Q}_l and \tilde{Q}_u represent the (linguistic) quantifiers “most” and “some”, respectively. Then the degree of membership of an element y to the lower approximation of A is equal to the truth value of the statement “Most elements similar to y are in A ”. The degree of membership of y to the upper approximation is equal to the truth value of the statement “Some elements are similar to y and are in A ”.

A. Examples of FQFRS models

1) $(\tilde{Z}_\Lambda^2, \tilde{Z}_\Upsilon)$ -FRS: let us take a look at the model derived from the most simple quantification model, the one from Zadeh. Let Λ and Υ be two RIM-quantifiers, then the lower and upper approximation for $(\tilde{Z}_\Lambda^2, \tilde{Z}_\Upsilon)$ -fuzzy rough sets are defined as:

$$\begin{aligned}(\underline{\text{apr}}_{R, \Lambda} A)(y) &:= \tilde{Z}_\Lambda^2(Ry, A) = \Lambda\left(\frac{|Ry \cap A|}{|Ry|}\right), \\ (\overline{\text{apr}}_{R, \Upsilon} A)(y) &:= \tilde{Z}_\Upsilon(Ry \cap A) = \Upsilon\left(\frac{|Ry \cap A|}{|X|}\right).\end{aligned}$$

This closely resembles the Vaguely Quantified Fuzzy Rough Sets (VQFRS) model [7], which uses the following lower and upper approximations:

$$\begin{aligned}(\underline{\text{apr}}_{R, \Lambda}^{\text{VQFRS}} A)(y) &:= \Lambda\left(\frac{|Ry \cap A|}{|Ry|}\right) = \tilde{Z}_\Lambda^2(Ry, A), \\ (\overline{\text{apr}}_{R, \Upsilon}^{\text{VQFRS}} A)(y) &:= \Upsilon\left(\frac{|Ry \cap A|}{|Ry|}\right) = \tilde{Z}_\Upsilon^2(Ry, A).\end{aligned}$$

For both models the lower approximations are identical, but whereas for VQFRS the lower and upper approximation only differ in their used RIM-quantifier, $(\tilde{Z}_\Lambda^2, \tilde{Z}_\Upsilon)$ -FRS evaluates the upper approximations using Zadeh's unary quantifier \tilde{Z}_Υ . Comparing the upper approximations of these two models, we can see that VQFRS will always be larger ($|X| \geq |Ry|$ and Υ is a RIM-quantifier). In some cases the upper approximation of VQFRS might be too large. For example, as soon as an element does not have a similar element ($Ry = \emptyset$) it is in the upper approximation of any concept A , even if there are many elements that are not that similar to y (e.g. $(Ry)(x) = 0.01$) but are in A (and not many elements similar to y) it will be in the upper approximation (cf. next example). Thus if one wants to discard the outlying elements from the upper approximation, this is problematic. This happens to a lesser extent with $(\tilde{Z}_\Lambda^2, \tilde{Z}_\Upsilon)$ -FRS, but it is still susceptible to it due to the accumulative nature of the Σ -count, as the following example shows.

Example IV.1. Suppose there are 10 elements in A with a similarity of 0.1 to $y \notin A$ and the rest of the elements are not similar to y at all ($|Ry \cap A| = 1$ and $|Ry| = 2$), then the upper approximation would always be 1 in the VQFRS approach:

$$(\overline{apr}_{R,\Upsilon}^{VQFRS} A)(y) = \tilde{Z}_\Upsilon^2(Ry, A) = \Upsilon(0.5) := 1,$$

since Υ should represent "some". In $(\tilde{Z}_\Lambda^2, \tilde{Z}_\Upsilon)$ -FRS we get a less extreme result:

$$(\overline{apr}_{R,\Upsilon} A)(y) = \tilde{Z}_\Upsilon(Ry \cap A) = \Upsilon\left(\frac{1}{|X|}\right).$$

So in conclusion, with VQFRS the outliers will always belong more to the upper approximation than with $(\tilde{Z}_\Lambda^2, \tilde{Z}_\Upsilon)$ -FRS. Lastly we note that using the existential quantifier (i.e. $\Upsilon = \Lambda_\exists$) the two upper approximations are equivalent ($|Ry| \geq 1$ since R is reflexive).

2) $(\tilde{Y}_\Lambda^{\rightarrow}, \tilde{Y}_\Upsilon)$ -FRS: since Yager's model is generally accepted as a better model compared to Zadeh's, we now take a look at $(\tilde{Y}_\Lambda^{\rightarrow}, \tilde{Y}_\Upsilon)$ -fuzzy rough sets. As shown in Equations (17) and (18), $(\tilde{Y}_\Lambda^{\rightarrow}, \tilde{Y}_\Upsilon)$ -FRS corresponds to OWAFRS which is preferred over VQFRS [20]. So this justifies the improvement from a fuzzy quantifier perspective why OWAFRS are better than VQFRS. To justify even more why this is a good model we know from Corollary III.7 that when using the Kleene-Dienes implicator, OWAFRS are equal to $(\mathcal{F}_{owa}(Q_\Lambda^{\rightarrow}), \mathcal{F}_{owa}(Q_\Upsilon))$ -FRS. So OWAFRS use a DFS mechanism, which is known to be semantically sound.

3) Other FQFRS: a problem with OWAFRS from a fuzzy quantifier perspective is that it makes use of the semi-fuzzy quantifier Q_Λ^{\rightarrow} , which is not that intuitive. Therefore $(\mathcal{F}_{owa}(Q_\Lambda^2), \mathcal{F}_{owa}(Q_\Upsilon))$ -FRS are preferable, since they make more sense on crisp sets (semi-fuzzy quantifiers), and are thus better suited for explainability when used for classification e.g. "Most people similar to y are not able to pay off their mortgage". Instead of \mathcal{F}_{owa} other DFS could also be used like the M_{CX} model [3], which is known to be a standard

DFS with many good properties, or the \mathcal{F}^A a non-standard DFS [28]. For a compromise between semantical soundness and computational efficiency, one could also use the quantifier $\tilde{Y}_\Lambda^2, \tilde{W}_\Lambda^2$ or \tilde{W}_Λ^2 using another integral (Remark III.3).

Example IV.2. Evaluating " ΛRy are A " using Q_Λ^{\rightarrow} gives us:

$$\Lambda\left(\frac{|Ry \rightarrow A|}{|X|}\right) = \Lambda\left(\frac{|\neg Ry| + |Ry \cap A|}{|X|}\right).$$

Thus the smaller the cardinality of Ry , the more true the statement is. This is not really what one would expect, because this causes the lower approximation to be large. Let $y \notin A$ be an instance that is an outlier (Ry only contains y), then the membership of y to the lower approximation of A is always very high (regardless of A). Using Q_Λ^2 instead would not result in this problem.

So this example suggests that using quantifiers with Q_Λ^2 as underlying semi-fuzzy quantifier might be preferable.

4) Choquet-based fuzzy rough sets: Choquet-based fuzzy rough sets correspond to $(C_{\mu_l}^{\mathcal{I}}, C_{\mu_u})$ -FRS with the fuzzy quantifiers $C_{\mu_l}^{\mathcal{I}}, C_{\mu_u}$ being defined as

$$C_{\mu_l}^{\mathcal{I}}(A, B) := \int \mathcal{I}(A, B) d\mu_l,$$

$$C_{\mu_u}(A) := \int A d\mu_u,$$

for every implicator \mathcal{I} and monotone measures μ_l and μ_u . When the measures μ_u and μ_l are symmetric these quantifiers are quantitative (i.e. each element is regarded as the same) and reduce to Yager's quantifiers like mentioned before. Note that all quantifiers mentioned in this section up until this point are quantitative. The interesting part of CFRS, when compared to OWAFRS, was that it allowed non-symmetric measures. We will now take a look at this from a FQFRS perspective for the non-symmetric measures discussed in Section II-F.

- Fuzzy removal measure Equation (14):

If $\mu_l = \mu_\forall$, then the quantifier $C_{\mu_l}^{\mathcal{I}}$ represents "for all except (maybe) elements of O ", thus it can be seen as the quantifier $\tilde{Y}_{\Lambda_\forall}^{\mathcal{I}}$ but not regarding elements of O .

- WOVA measure Equation (15):

If $\mu_l = \mu_\forall$, then the quantifier $C_{\mu_l}^{\mathcal{I}}$ represents " Λ elements except (maybe) elements of O ", thus it can be seen as the quantifier $\tilde{Y}_\Lambda^{\mathcal{I}}$ but not regarding elements of O . Do note that for $\Lambda = \Lambda_\forall$ this quantifier represents the same quantifier as the fuzzy removal measure, but the evaluation is different.

B. Confidence-based FQFRS

In the previous section we have seen that we are able to seamlessly incorporate outlier information in FQFRS by making use of non-quantitative quantifiers (an implicit way). But it is also possible to do this using quantitative quantifiers (a more explicit way):

Definition IV.2. Given a reflexive fuzzy relation $R \in \mathcal{F}(X \times X)$, $O \in \mathcal{F}(X)$, fuzzy quantifiers $\tilde{Q}_l : (\tilde{P}(X))^3 \rightarrow [0, 1]$ and

$\tilde{Q}_u : (\tilde{P}(X))^2 \rightarrow [0, 1]$ and $A \in \mathcal{F}(X)$, then the lower and upper approximation of A w.r.t. R , \tilde{Q}_l and \tilde{Q}_u are given by:

$$\begin{aligned} (\underline{apr}_{R, \tilde{Q}_l} A)(y) &= \tilde{Q}_l(O, Ry, A), \\ (\overline{apr}_{R, \tilde{Q}_u} A)(y) &= \tilde{Q}_u(O, Ry \cap_C A). \end{aligned}$$

For example, let CS be a fuzzy set describing the accuracy-/confidence of the instances in X , \tilde{Q}_l a quantifier modelling “Most elements except maybe non confident elements” and \tilde{Q}_u a quantifier modelling “Some confident elements”, then an element y is “in” the lower approximation if most accurate/confident elements indiscernible from y are in A , and an element y is “in” the upper approximation if there are some accurate/confident elements that are indiscernible to y and are “in” A .

Example IV.3. Using a binary quantifier \tilde{Q}_l (e.g. representing “Most”) and a unary quantifier \tilde{Q}_u (e.g. representing “Some”) we can do this as follows:

$$\begin{aligned} (\underline{apr}_{R, \tilde{Q}_l} A)(y) &= \tilde{Q}_l((\tilde{\neg}O) \cap_C Ry, A), \\ (\overline{apr}_{R, \tilde{Q}_u} A)(y) &= \tilde{Q}_u((\tilde{\neg}O) \cap_C Ry \cap_C A). \end{aligned}$$

V. CONCLUSION

We have introduced *fuzzy quantifier-based fuzzy rough sets* (FQFRS), a general definition of fuzzy rough sets based on fuzzy quantifiers. FQFRS allows to position existing models and compare them on the basis of their associated fuzzy quantifiers. In addition, this general model can lead to improved models in terms of performance and interpretability. Currently, there are only a few models that make explicit use of quantifiers, but these can be improved by using more semantically sound evaluation models. Furthermore, we have introduced novel binary quantification models based on integrals, that might give us a good compromise between computational efficiency and semantical soundness. Finally, we have introduced confidence-based FQFRS that are able to perform active outlier/noise reduction, i.e., taking into account outlier information (e.g., obtained by an outlier detection algorithm), in a more explicit and general way compared to CFRS.

VI. FUTURE WORK

From a theoretical perspective, it is interesting to find out how the properties of the used quantifiers translate to properties of the corresponding fuzzy rough sets, and vice versa. Also, it is possible to study the new fuzzy quantifiers including the quantifiers corresponding with the fuzzy removal and WOWA measures, and comparing them with existing quantifiers. Lastly we plan to perform an experimental study of the performance of the different fuzzy quantifier-based fuzzy rough sets by testing it in fuzzy rough set based classifiers similar to those considered by Lenz et al [29].

ACKNOWLEDGMENT

The research reported in this paper was conducted with the financial support of the Odysseus programme of the Research Foundation – Flanders (FWO). The grant number is G0H9118N.

REFERENCES

- [1] L. A. Zadeh, “A computational approach to fuzzy quantifiers in natural languages,” in *Computational linguistics*. Elsevier, 1983, pp. 149–184.
- [2] R. R. Yager, “Quantifier guided aggregation using owa operators,” *International Journal of Intelligent Systems*, vol. 11, no. 1, pp. 49–73, 1996.
- [3] I. Glöckner, *Fuzzy quantifiers: a computational theory*. Springer, 2008, vol. 193.
- [4] Z. Pawlak, “Rough sets,” *International journal of computer & information sciences*, vol. 11, no. 5, pp. 341–356, 1982.
- [5] D. Dubois and H. Prade, “Rough fuzzy sets and fuzzy rough sets,” *International Journal of General System*, vol. 17, no. 2-3, pp. 191–209, 1990.
- [6] S. Vluymans, L. D’eer, Y. Saeys, and C. Cornelis, “Applications of fuzzy rough set theory in machine learning: a survey,” *Fundamenta Informaticae*, vol. 142, no. 1-4, pp. 53–86, 2015.
- [7] C. Cornelis, M. De Cock, and A. M. Radzikowska, “Vaguely quantified rough sets,” in *International Workshop on Rough Sets, Fuzzy Sets, Data Mining, and Granular-Soft Computing*. Springer, 2007, pp. 87–94.
- [8] J. Fernández Salido and S. Murakami, “On β -precision aggregation,” *Fuzzy Sets and Systems*, vol. 139, no. 3, pp. 547–558, 2003. doi: [https://doi.org/10.1016/S0165-0114\(03\)00003-4](https://doi.org/10.1016/S0165-0114(03)00003-4). [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0165011403000034>
- [9] —, “Rough set analysis of a general type of fuzzy data using transitive aggregations of fuzzy similarity relations,” *Fuzzy Sets and Systems*, vol. 139, no. 3, pp. 635–660, 2003. doi: [https://doi.org/10.1016/S0165-0114\(03\)00124-6](https://doi.org/10.1016/S0165-0114(03)00124-6). [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0165011403001246>
- [10] A. Mieszkowicz-Rolka and L. Rolka, “Variable precision fuzzy rough sets,” in *Transactions on Rough Sets I*. Springer, 2004, pp. 144–160.
- [11] Y. Yao, J. Mi, and Z. Li, “A novel variable precision (θ, σ) -fuzzy rough set model based on fuzzy granules,” *Fuzzy Sets and Systems*, vol. 236, pp. 58–72, 2014. doi: <https://doi.org/10.1016/j.fss.2013.06.012> Theme: Algebraic Aspects of Fuzzy Sets. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0165011413002753>
- [12] Q. Hu, S. An, and D. Yu, “Soft fuzzy rough sets for robust feature evaluation and selection,” *Information Sciences*, vol. 180, no. 22, pp. 4384–4400, 2010. doi: <https://doi.org/10.1016/j.ins.2010.07.010>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0020025510003282>
- [13] A. Hadrani, K. Guennoun, R. Saadane, and M. Wahbi, “Fuzzy rough sets: Survey and proposal of an enhanced knowledge representation model based on automatic noisy sample detection,” *Cognitive Systems Research*, vol. 64, pp. 37–56, 2020. doi: <https://doi.org/10.1016/j.cogsys.2020.05.001>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1389041720300255>
- [14] S. An, Q. Hu, W. Pedrycz, P. Zhu, and E. C. C. Tsang, “Data-distribution-aware fuzzy rough set model and its application to robust classification,” *IEEE Transactions on Cybernetics*, vol. 46, no. 12, pp. 3073–3085, 2016. doi: 10.1109/TCYB.2015.2496425
- [15] S. An, Q. Hu, and C. Wang, “Probability granular distance-based fuzzy rough set model,” *Applied Soft Computing*, vol. 102, p. 107064, 2021. doi: <https://doi.org/10.1016/j.asoc.2020.107064>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1568494620310024>
- [16] C. Cornelis, N. Verbiest, and R. Jensen, “Ordered weighted average based fuzzy rough sets,” in *International Conference on Rough Sets and Knowledge Technology*. Springer, 2010, pp. 78–85.
- [17] A. Theerens, O. U. Lenz, and C. Cornelis, “Choquet-based fuzzy rough sets,” *International Journal of Approximate Reasoning*, 2022. doi: 10.1016/j.ijar.2022.04.006
- [18] H.-P. Kriegel, P. Kroger, E. Schubert, and A. Zimek, “Interpreting and unifying outlier scores,” in *Proceedings of the 2011 SIAM International Conference on Data Mining*. SIAM, 2011, pp. 13–24.
- [19] L. A. Zadeh, “Fuzzy sets,” *Information and Control*, 1965.
- [20] L. D’eer, N. Verbiest, C. Cornelis, and L. Godo, “A comprehensive study of implicator–conjunctive-based and noise-tolerant fuzzy rough sets: definitions, properties and robustness analysis,” *Fuzzy Sets and Systems*, vol. 275, pp. 1–38, 2015.
- [21] R. R. Yager, “On ordered weighted averaging aggregation operators in multicriteria decisionmaking,” *IEEE Transactions on systems, Man, and Cybernetics*, vol. 18, no. 1, pp. 183–190, 1988.

- [22] G. Beliakov, A. Pradera, T. Calvo *et al.*, *Aggregation functions: A guide for practitioners*. Springer, 2007, vol. 221.
- [23] Z. Wang and G. J. Klir, *Generalized measure theory*. Springer Science & Business Media, 2010, vol. 25.
- [24] A. Cascallar-Fuentes, A. Ramos-Soto, and A. Bugarín-Diz, “An experimental study on the behaviour of fuzzy quantification models,” in *ECAI 2020*. IOS Press, 2020, pp. 267–274.
- [25] M. Delgado, M. D. Ruiz, D. Sánchez, and M. A. Vila, “Fuzzy quantification: a state of the art,” *Fuzzy Sets and Systems*, vol. 242, pp. 1–30, 2014.
- [26] V. Torra, “The weighted owa operator,” *International Journal of Intelligent Systems*, vol. 12, no. 2, pp. 153–166, 1997.
- [27] —, “On some relationships between the wowa operator and the Choquet integral,” in *Proceedings of the IPMU 1998 Conference, Paris, France*. Citeseer, 1998, pp. 818–824.
- [28] F. Diaz-Hermida, D. Losada, A. Bugarin, and S. Barro, “A probabilistic quantifier fuzzification mechanism: The model and its evaluation for information retrieval,” *IEEE Transactions on Fuzzy Systems*, vol. 13, no. 5, pp. 688–700, 2005. doi: 10.1109/TFUZZ.2005.856557
- [29] O. U. Lenz, D. Peralta, and C. Cornelis, “Scalable approximate frnn-owa classification,” *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 5, pp. 929–938, 2019. doi: 10.1109/TFUZZ.2019.2949769

Feature Selection and Ranking Method based on Intuitionistic Fuzzy Matrix and Rough Sets

Bich Khue Vo
University of Finance and Marketing
Ho Chi Minh City, Viet Nam
Email: votbkhue@gmail.com
and vokhue@ufm.edu.vn

Hung Son Nguyen
Institute of Computer Science
University of Warsaw
ul. Banacha 2,02-927 Warszawa, Poland
Email: son@mimuw.edu.pl

Abstract—In this paper we propose a novel rough-fuzzy hybridization technique to feature selection and feature ranking problem. The idea is to model the local preference relation between pair of features by intuitionistic fuzzy values and search for a feature ranking that is consistent with those constraints. We apply the techniques used in group decision making where constraints are presented in form of intuitionistic fuzzy preference relation. The proposed method has been illustrated by some simple examples and verified on a benchmark dataset.

Index Terms—Intuitionistic fuzzy matrix, feature ranking, reducts, rough sets.

I. INTRODUCTION

DECISION making plays an extremely important role in the real life problems. With strong development of datasets having a great deal of features, recognizing preferred features is a challenging task. In recent years, rough-fuzzy hybridization becomes a hot trend with great success in machine learning, data mining, decision making, etc. [1].

Usually, a group decision-making process must collect all decision-makers's opinions, establish a suitable method for measuring them, obtain the final scores of all alternatives, and then rank them. Decision makers have to rank the alternatives, find the most preferred feature in order to make decision. So, preference relations are effective techniques to gather the overviews of a group of decision makers. Recently, many researchers have developed many methods of preference relations [2], [3], [4], [5], [6][7]. To solve decision-making problems with uncertain information or not precise judgments, preference relations with Zadeh's fuzzy sets are proposed. There are two number values to measure the degree of membership and nonmembership in Zadeh's fuzzy theory with sum is one, but ignore the decision maker's hesitation in the decision making process. The Atanassov's intuitionistic fuzzy sets consider fully expressing affirmation, negation and hesitation. Particularly, the usage of intuitionistic fuzzy preference relation to affirmative, negative and hesitant characteristics makes the research problem more and more attractive and competitive.

Real world problems may have numerous irrelevant features, and in such cases feature selection can help decision makers choose important information hidden in the full dataset. Feature selection (FS) method belongs to one of the

three main groups: embedded, filter or wrapper methods and can be defined as selecting a subset of available features in a dataset that is associated with the response variables by excluding irrelevant and unnecessary features. An alternative to FS for dimensionality reduction is feature extraction (FE) in which original features are united and then projected into a new feature space with smaller dimensionality. In this paper, we interpret the feature selection and feature ranking as decision making problems and apply the recent techniques for solving it.

In the process of group making decisions, there are often many different opinions of defining the degree of certainty, uncertainty or hesitation among decision makers (DMs). Therefore, it's necessary to define an intuitionistic fuzzy preference relation which have the capability of representing all selections. The consistency of intuitionistic fuzzy preference relations (IFPRs) and the priority weights of DMs gathered from these preference relations play a vital role in group making decision to lead to the most best result. In some works, the consequences of additive consistent and multiplicative consistent IFPRs on priority weights is examined and considered to calculate the priority weights. Numerical analyses have shown that the ranking of the individual priority weights do not differ seriously despite of the different priority weight vectors of the individual priority weights. The intuitionistic fuzzy preference relation is introduced as an indispensable tool for enabling decision-makers to judge the superiority or inferiority of one object to another, in the presence of fuzziness. Ranking methods for alternatives with intuitionistic fuzzy information are expressed straightforwardly and efficiently to get the solution of group decision problems.

The paper is organized as follows: in Section II we recall some basic notions in intuitionistic fuzzy set theory and rough set theory. Section III describes ranking methods that are consistent or semi-consistent with one or more intuitionistic fuzzy preference relations. In Section IV, we present a rough-fuzzy hybridization method for feature ranking and illustrate the proposed method on the base of some simple examples. The results of experiments on the accuracy of the feature ranking methods in the context of classification task are reported in Section V. The conclusions and plan for future research are presented in Section VI.

II. BASIC NOTIONS

In this section we present some fundamental knowledge about intuitionistic fuzzy set, intuitionistic fuzzy relation and feature reduction in rough set theory.

A. Intuitionistic fuzzy sets and Intuitionistic fuzzy matrices

Fuzzy set (FS) theory which was introduced by Zadeh in 1975 [8] just considers the problems with the degree of membership and non-membership without mentioning the degree of hesitation of no decision-making. The Atanassov's intuitionistic fuzzy set (IFS) theory [9] considers fully expressing affirmation, negation and hesitation of decision-makers. Therefore, with real-life situations, IFS theory solves the problems more successfully than FS theory. In this part, some basic notions related to IFS are recalled.

Definition 1 (IFS [9]). Let $X = \{x_1, x_2, \dots, x_m\}$ be a finite universal set. An intuitionistic fuzzy set in X is a set $A = \{(x_i, \mu_A(x_i), \nu_A(x_i)) : x_i \in X\}$, where $\mu_A(x_i), \nu_A(x_i) \in [0, 1]$ and $0 \leq \mu_A(x_i) + \nu_A(x_i) \leq 1$ for any $x_i \in X$. The functions $\mu_A : X \rightarrow [0, 1]$ and $\nu_A : X \rightarrow [0, 1]$ are called the membership and non-membership functions, respectively.

The value $\pi_A(x_i) = 1 - \mu_A(x_i) - \nu_A(x_i)$ is called the intuitionistic index of the element x_i in the set A . It describes a degree of hesitation (or uncertainty) whether x_i is in A or not. For any $x_i \in X$, we have $0 \leq \pi_A(x_i) \leq 1$.

The class of IFS in a universe X is denoted by $\mathcal{IFS}(X)$.

Let \mathcal{F} be the set of tuples (a_1, a_2) , where $a_1, a_2 \in [0, 1]$ and $a_1 + a_2 \leq 1$, i.e.,

$$\mathcal{F} = \{(a_1, a_2) \in [0, 1]^2 : a_1 + a_2 \leq 1\}.$$

A partial order $\leq_{\mathcal{F}}$ over \mathcal{F} defined by:

$$(a_1, a_2) \leq_{\mathcal{F}} (b_1, b_2) \Leftrightarrow a_1 \leq b_1 \text{ and } a_2 \geq b_2.$$

The elements of \mathcal{F} are called the intuitionistic fuzzy values (IFV), of which $(0, 1)$ is the least element and $(1, 0)$ is the greatest element. The operations in $(\mathcal{F}, \leq_{\mathcal{F}})$ are defined by

$$(a_1, a_2) \vee (b_1, b_2) = (\max\{a_1, b_1\}, \min\{a_2, b_2\}).$$

$$(a_1, a_2) \wedge (b_1, b_2) = (\min\{a_1, b_1\}, \max\{a_2, b_2\})$$

$$(a_1, a_2)^c = (a_2, a_1).$$

Each IFS of a universe X is in fact a map from X to \mathcal{F} . If A and B are two IFSs defined by (μ_A, ν_A) and (μ_B, ν_B) , correspondingly, then the union, intersection and complement are defined as follows:

$$\begin{aligned} (\mu_{A \cup B}, \nu_{A \cup B}) &= (\mu_A, \nu_A) \vee (\mu_B, \nu_B) \\ &= (\max\{\mu_A, \mu_B\}, \min\{\nu_A, \nu_B\}) \end{aligned}$$

$$\begin{aligned} (\mu_{A \cap B}, \nu_{A \cap B}) &= (\mu_A, \nu_A) \wedge (\mu_B, \nu_B) \\ &= (\min\{\mu_A, \mu_B\}, \max\{\nu_A, \nu_B\}) \end{aligned}$$

$$(\mu_{A^c}, \nu_{A^c}) = (\mu_A, \nu_A)^c = (\nu_A, \mu_A)$$

$$A \subseteq B \Leftrightarrow (\mu_A, \nu_A) \leq_{\mathcal{F}} (\mu_B, \nu_B)$$

Relations (between two sets X and Y) in traditional set theory are defined as subsets of the Cartesian product $X \times Y$.

It is quite natural to define intuitionistic fuzzy relations as IFSs in $X \times Y$. If $X = \{x_1, \dots, x_m\}$ and $Y = \{y_1, \dots, y_n\}$, then any intuitionistic fuzzy relation in $X \times Y$ can be represented by an $m \times n$ matrix $R = (\rho_{ij})_{m \times n}$, where $\rho_{ij} = (\mu_{ij}, \nu_{ij}) \in \mathcal{F}$ is the IFV describing the membership and non-membership of (x_i, y_j) to this relation.

Definition 2 (Intuitionistic fuzzy matrices - IFM). Any matrix P of order $n \times m$ with values from $\mathcal{F} = \{(a_1, a_2) \in [0, 1]^2 : a_1 + a_2 \leq 1\}$ is called Intuitionistic Fuzzy Matrices. An IFM is said to be square intuitionistic fuzzy matrix (SIFM) if the number of rows is equal to the number of columns. Moreover:

- 1) An identity IFM \mathbb{I} of order n is the square intuitionistic fuzzy matrix (SIFM) of order n with all diagonal entries $(1, 0)$ and non-diagonal entries $(0, 1)$.
- 2) A null intuitionistic fuzzy matrix (IFM) \mathbb{O} of order n is the square intuitionistic fuzzy matrix (SIFM) of order n with all entries $(0, 1)$.

The concepts of intuitionistic fuzzy relation and intuitionistic fuzzy matrix (IFM) have been studied by many authors [10], [11], [12]. IFM is a generalization of Fuzzy Matrix and has been useful in dealing with decision-making, clustering analysis, relational equations, etc.

Since IFM is an extension of FM by replacing the values from $[0, 1]$ by IFV, i.e. elements of $\mathcal{F} = \{(a_1, a_2) \in [0, 1]^2 : a_1 + a_2 \leq 1\}$, and the fuzzy operations \vee and \wedge were extended for the elements of \mathcal{F} . Most of operations on fuzzy matrices can be also extended for IFMs. In particular, if $A = (a_{ij}), B = (b_{ij}) \in \mathcal{F}_{m \times n}$, where $a_{ij} = (\mu_{ij}^a, \nu_{ij}^a), b_{ij} = (\mu_{ij}^b, \nu_{ij}^b)$, for $i = 1, \dots, m; j = 1, \dots, n$ then

- disjunction and conjunction are defined by:

$$A \vee B = (a_{ij} \vee b_{ij})_{m \times n} = (\delta_{ij})_{m \times n}$$

$$\text{where } \delta_{ij} = (\min\{\mu_{ij}^a, \mu_{ij}^b\}, \max\{\nu_{ij}^a, \nu_{ij}^b\}).$$

$$A \wedge B = (a_{ij} \wedge b_{ij})_{m \times n} = (\gamma_{ij})_{m \times n}$$

$$\text{where } \gamma_{ij} = (\max\{\mu_{ij}^a, \mu_{ij}^b\}, \min\{\nu_{ij}^a, \nu_{ij}^b\}).$$

- Comparison: $A \leq B \Leftrightarrow a_{ij} \leq b_{ij}$ for all $i = 1, 2, \dots, m$ and $j = 1, 2, \dots, n$.
- The tranpose of $A = (a_{ij})_{m \times n}$ is $A^T = (x_{ij})_{n \times m}$, where $x_{ij} = a_{ji}$.
- The composition of two relations or the product of two matrices $A \in \mathcal{F}_{m \times n}$ and $C \in \mathcal{F}_{n \times l}$ is the matrix $D = (d_{ij}) \in \mathcal{F}_{m \times l}$, where

$$\begin{aligned} d_{ij} &= \bigvee_{k=1}^n (a_{ik} \wedge c_{kj}) = \left(\bigvee_{k=1}^n (\mu_{ik}^a \wedge \mu_{kj}^c), \bigwedge_{k=1}^n (\nu_{ik}^a \vee \nu_{kj}^c) \right) \\ &= \left(\max_{k=1, \dots, n} \{\min\{\mu_{ik}^a, \mu_{kj}^c\}\}, \min_{k=1, \dots, n} \{\max\{\nu_{ik}^a, \nu_{kj}^c\}\} \right). \end{aligned}$$

This operation is denoted by $D = A \circ C$.

B. Feature selection and feature ranking problem

In machine learning, a classification task is defined as the problem of learning the partition of a set of objects into subsets called decision classes (or briefly classes). The partition should

be expressed in terms of object features. Let U be a set of objects. Any function $a : U \rightarrow V_a$, where V_a is the domain (the set of possible values) of a , is called a feature or attribute for U . If a is a measurement such as a person's weight, height, blood pressure or the weather temperature, i.e. V_a is a real interval, then a is called the numeric or quantitative feature. Otherwise, if the values in V_a are not comparable, or if they can not be ordered in a linear order, then a is called categorical, symbolic or qualitative feature.

Decision table is a tuple $T = (U, A \cup \{d\})$, where $U = \{u_1, u_2, \dots, u_n\}$ is a finite set of objects, $A = \{a_1, a_2, \dots, a_m\}$ is a finite set of features called conditional attributes and d is the decision attribute, i.e. the attribute defining the partition of objects into decision classes. Any decision table $T = (U, A \cup \{d\})$ stores a training data set $D = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n)\}$ for machine learning algorithms, where $\mathbf{x}_i = (a_1(u_i), \dots, a_m(u_i))$ and $y_i = d(u_i)$.

The goal of the feature selection (FS) problem for decision table $T = (U, A \cup \{d\})$ is to determine the minimal subsets of A satisfying a particular classification performance requirement. Feature ranking problem is to order the feature in a ranking list so that the more important features are at the beginning of the ranking list while the less important features are at the end of the ranking list.

C. Rough Sets and feature selection problem

In rough set theory, the FS problem is formulated as a problem of searching for reducts. Intuitively, reducts of a decision table $T = (U, A \cup \{d\})$ are the minimal subsets (with respect to inclusion) of the set of all attributes A that guarantee the classification performance of the reduced decision table. In the pioneering paper in rough set theory [13], [14], [15], the classification performance was defined in terms of the discernibility between the objects. Formally, for any subset $B \subset A$ and two objects $u, v \in U$, we say that

$$\begin{aligned} B \text{ discerns } u \text{ and } v \text{ (or } u, v \text{ are discernible by } B) \\ \Leftrightarrow \text{there exists } b \in B \text{ such that } b(u) \neq b(v) \end{aligned}$$

The set $B \subset A$ is called the reduct of decision table T if

- For any $u, v \in U$, if u, v are discernible by A and discernible by $\{d\}$ then u, v are discernible by B ;
- No proper subset of B satisfies the previous condition.

This original concept of reducts has been generated by many researchers [16], [17], [18]. However, the solution space for the problem of searching for the reduct with minimal cardinality is $2^m - 1$ where $m = |A|$.

The first approach to minimal reduct problem has been proposed in [14]. For the given decision table $T = (U, A \cup \{d\})$ with $U = \{u_1, u_2, \dots, u_n\}$ and $A = \{a_1, a_2, \dots, a_m\}$, the authors constructed a discernibility matrix $\mathbb{M}(T)$, which is in fact a function $\mathbb{M}(T) : U \times U \rightarrow \mathcal{P}(A)$, where $\mathcal{P}(A)$ denotes the power set of A . For each two objects $u_i, u_j \in U$, we denote $\mathbb{M}(T)(u_i, u_j) = S_{ij}$, where

$$S_{ij} = \begin{cases} \emptyset & \text{if } d(x_i) = d(x_j) \\ \{a \in A : a(x_i) \neq a(x_j)\} & \text{otherwise} \end{cases}$$

The example of discernibility matrix is presented in Table II.

One can notice that a subset of attributes $B \subset A$ discerns a pair of objects $u_i, u_j \in U$ if and only if $B \cap S_{ij} \neq \emptyset$.

Let us notice that for any $u_i, u_j \in U$ we have $S_{ij} = S_{ji}$. That's why, in case of the decision table with two decision classes, the discernibility matrix can be simplified into $p \times q$ matrix where p and q are the cardinalities of the two decision classes.

Since minimal reduct calculation problem is NP-hard, many heuristics algorithms are using random permutations of features [16], [19] as a nondeterministic policy and the algorithm is searching for the reduct according to the attribute order defined by the given permutations.

III. FROM INTUITIONISTIC FUZZY PREFERENCE RELATION TO RANKING.

In this Section we present a method of using Intuitionistic Fuzzy Sets to approximate the concept of fuzzy preference [20] [21], [22].

Definition 3 (intuitionistic fuzzy preference relation). *An intuitionistic fuzzy preference relation B on $X = \{x_1, \dots, x_n\}$ is defined as a matrix $B = (b_{ij})_{n \times n}$, where $b_{ij} = (\mu_{ij}, \nu_{ij})$ for all $i, j = 1, 2, \dots, n$ is an intuitionistic fuzzy value, composed by the certainty degree μ_{ij} to which x_i is preferred to x_j and the certainty degree ν_{ij} to which x_i is non-preferred to x_j , and $\pi_{ij} = 1 - \mu_{ij} - \nu_{ij}$ is interpreted as the hesitation degree to which x_i is preferred to x_j . Moreover, μ_{ij} and ν_{ij} satisfy the following conditions:*

$$\mu_{ij} + \nu_{ij} \leq 1, \quad \mu_{ij} = \nu_{ji}, \quad \mu_{ii} = \nu_{ii} = 0.5$$

for all $i, j = 1; 2; \dots, n$.

Usually, the intuitionistic fuzzy preference relation expresses the opinions of the decision makers about each pair of choices (alternatives), but we would like to convert this relation into a linear order (a ranking list). We can do it by assigning a weight w_i to the i -th choice so that the higher weight means the more preferred choice. Without loss of generality, we can assume that the weight vector can be determined in form of a probability vector, i.e. a vector $\mathbf{w} = (w_1, w_2, \dots, w_n)^T$ such that $w_i \in [0, 1]$ for $i = 1, \dots, n$ and $\sum_{i=1}^n w_i = 1$. We can define the concept of consistent preference relation as follows:

Definition 4 (Additive consistent preference relation). *An intuitionistic fuzzy preference relation $B = ((\mu_{ij}, \nu_{ij}))_{n \times n}$ on $X = \{x_1, \dots, x_n\}$ is an additive consistent preference relation if there exists a probability vector $\mathbf{w} = (w_1, w_2, \dots, w_n)^T$ satisfying the condition:*

$$\mu_{ij} \leq 0.5(w_i - w_j + 1) \leq 1 - \nu_{ij} \quad (1)$$

for all $1 \leq i < j \leq n$.

It is obvious that not every intuitionistic fuzzy preference relation is also an additive consistent relation. In this case the

condition in Eq. (1) can be relaxed by introducing the non-negative deviation variables l_{ij} and r_{ij} for $1 \leq i < j \leq n$ such that

$$\mu_{ij} - l_{ij} \leq 0.5(w_i - w_j + 1) \leq 1 - \nu_{ij} + r_{ij} \quad (2)$$

for all $1 \leq i < j \leq n$. As the deviation variables l_{ij} and r_{ij} become smaller, B becomes closer to an additive consistent intuitionistic fuzzy preference relation. Therefore, in order to find the smallest deviation variables one can developed the following linear optimization model [23], [21]:

Model (A1):

$$\delta = \min \sum_{i=1}^{n-1} \sum_{j=i+1}^n (l_{ij} + r_{ij})$$

$$s.t. \left\{ \begin{array}{l} 0.5(w_i - w_j + 1) + l_{ij} \geq \mu_{ij} \\ 0.5(w_i - w_j + 1) - r_{ij} \leq 1 - \nu_{ij} \\ l_{ij}, r_{ij} \geq 0 \\ w_i \geq 0 \quad \text{for } i = 1, \dots, n, \\ \sum_{i=1}^n w_i = 1 \end{array} \right\} \quad (*)$$

where (*) must be true for all $1 \leq i < j \leq n$.

Let δ^o be the optimal value and let l_{ij}^o and r_{ij}^o for $1 \leq i < j \leq n$ be optimal deviation values of the optimization model (A1). One can see that if $\delta^o = 0$ then B is an additive consistent intuitionistic fuzzy preference relation. Otherwise, we can improve the additive consistency of B by defining the new intuitionistic fuzzy preference relation $\hat{B} = ((\hat{\mu}_{ij}, \hat{\nu}_{ij}))_{n \times n}$, where

$$\hat{\mu}_{ij} = \begin{cases} \mu_{ij} - l_{ij}^o & \text{if } i < j \\ 0.5 & \text{if } i = j \\ \hat{\nu}_{ij} & \text{if } i > j \end{cases} \quad \hat{\nu}_{ij} = \begin{cases} \nu_{ij} - r_{ij}^o & \text{if } i < j \\ \nu_{ij} = 0.5 & \text{if } i = j \\ \hat{\mu}_{ij} & \text{if } i > j \end{cases}$$

Based on matrix \hat{B} we can calculate the priority weight vector $\mathbf{w} = (w_1, \dots, w_n)^T$ by establishing the weight intervals $[w_k^-, w_k^+]$ for each $k = 1, \dots, n$. In order to do that, we solve the following optimization models

Model (A2): for each $k = 1, 2, \dots, n$:

$$(w_k^-, w_k^+) = (\min w_k, \max w_k)$$

$$s.t. \left\{ \begin{array}{l} 0.5(w_i - w_j + 1) \geq \hat{\mu}_{ij} \\ 0.5(w_i - w_j + 1) \leq 1 - \hat{\nu}_{ij} \\ w_i \geq 0 \quad \text{for } i = 1, \dots, n, \\ \sum_{j=1}^n w_j = 1. \end{array} \right\} \quad (*)$$

where (*) must be true for all $1 \leq i < j \leq n$.

It has been shown [23] that if \hat{B} is additive consistent then Model (A2) will return an unique solution for the considered optimization problem.

Definition 5 (Multiplicative consistent preference relation:). *An intuitionistic fuzzy preference relation $B = ((\mu_{ij}, \nu_{ij}))_{n \times n}$*

on $X = \{x_1, \dots, x_n\}$ is an multiplicative consistent preference relation if there exists a probability vector $\mathbf{w} = (w_1, w_2, \dots, w_n)^T$ satisfying the condition:

$$\mu_{ij} \leq \frac{w_i}{w_i + w_j} \leq 1 - \nu_{ij} \text{ for all } 1 \leq i < j \leq n. \quad (3)$$

Checking for the multiplicative consistency is quite similar to the additive consistency. In this case, we can establish the optimization model (M1). In contrast to model (A1), this model is nonlinear.

Model (M1):

$$\delta = \min \sum_{i=1}^{n-1} \sum_{j=i+1}^n (l_{ij} + r_{ij})$$

$$s.t. \left\{ \begin{array}{l} \frac{w_i}{w_i + w_j} + l_{ij} \geq \mu_{ij} \\ \frac{w_i}{w_i + w_j} - r_{ij} \leq 1 - \nu_{ij} \\ l_{ij}, r_{ij} \geq 0 \\ w_i \geq 0 \quad \text{for } i = 1, \dots, n, \\ \sum_{i=1}^n w_i = 1 \end{array} \right\} \quad (*)$$

where (*) must be true for all $1 \leq i < j \leq n$.

Let δ^* be the optimal value and let l_{ij}^* and r_{ij}^* for $1 \leq i < j \leq n$ be optimal deviation values of the optimization model (M1). One can see that if $\delta^* = 0$ then B is an multiplicative consistent intuitionistic fuzzy preference relation. Otherwise, we can improve the multiplicative consistency of B by defining the new intuitionistic fuzzy preference relation $B^* = ((\mu_{ij}^*, \nu_{ij}^*))_{n \times n}$, where

$$\mu_{ij}^* = \begin{cases} \mu_{ij} - l_{ij}^* & \text{if } i < j \\ 0.5 & \text{if } i = j \\ \nu_{ij}^* & \text{if } i > j \end{cases} \quad \nu_{ij}^* = \begin{cases} \nu_{ij} - r_{ij}^* & \text{if } i < j \\ 0.5 & \text{if } i = j \\ \mu_{ij}^* & \text{if } i > j \end{cases}$$

Based on matrix B^* we can calculate the priority weight vector $\mathbf{w} = (w_1, \dots, w_n)^T$ by establishing the weight intervals $[w_k^-, w_k^+]$ for each $k = 1, \dots, n$. In order to do that, we solve the following optimization models

Model (M2): for each $k = 1, 2, \dots, n$:

$$(w_k^-, w_k^+) = (\min w_k, \max w_k)$$

$$s.t. \left\{ \begin{array}{l} \frac{w_i}{w_i + w_j} \geq \mu_{ij}^* \\ \frac{w_i}{w_i + w_j} \leq 1 - \nu_{ij}^* \\ w_i \geq 0 \quad \text{for } i = 1, \dots, n, \\ \sum_{j=1}^n w_j = 1. \end{array} \right\} \quad (*)$$

where (*) must be true for all $1 \leq i < j \leq n$.

IV. HYBRID METHOD FOR FEATURE RANKING PROBLEM

In this section we present a rough-fuzzy hybridization technique for searching for the optimal ranking list of features, called RAFAR (Rough-fuzzy Algorithm For Attribute Ranking). We introduce the concept of fuzzy discernibility matrix,

which is a generalization of discernibility matrix in rough set theory, and combine it with the ranking calculation methods from intuitionistic fuzzy preference relations.

A. Construction of IFPR from decision table

The general framework of our proposition is presented in the Fig. 1:

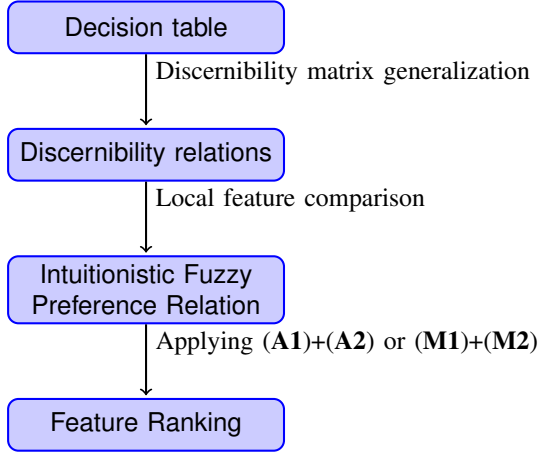


Fig. 1: The general framework of our proposition.

For a given decision table $T = (U, A \cup \{d\})$, where $U = \{u_1, u_2, \dots, u_n\}$ and $A = \{a_1, a_2, \dots, a_m\}$. To simplify the description, let us assume that d is a binary decision attribute, e.g., $V_d = \{-1, 1\}$. The proposed methods also work for the multi-class case.

Let $\mathcal{RED}(T) = \{R \in A : R \text{ is a reduct in } T\}$ denotes the set of all reducts of the decision table T . In [24], the authors classified the attributes into 3 categories:

- Core attributes: the attributes that occur in all reducts:

$$CORE(T) = \bigcap_{R \in \mathcal{RED}} R$$

- Reductive attributes: the attributes that present in at least one reduct:

$$REAT(T) = A - \bigcup_{R \in \mathcal{RED}} R$$

- The attribute is called redundant if it is not a reductive attribute.

Our aim is to generate a feature ranking that at least follows this classification.

Let consider the case of decision table with symbolic values. We define the two decision classes by

$$U_- = \{u \in U : d(u) = -1\}; \quad U_+ = \{u \in U : d(u) = 1\}$$

For each feature $a_k \in A$, we define a function $P_{a_k} : U_+ \times U_- \rightarrow \{0, 1\}$ by

$$P_{a_k}(u_i, u_j) = \begin{cases} 1 & \text{if } a(u_i) \neq a(u_j), \\ 0 & \text{otherwise.} \end{cases}$$

We can see that if a_k is a symbolic feature then P_{a_k} is a relation between U_- and U_+ in the traditional meaning. This relation is called the discernibility relation [14]. Moreover, if $\mathbb{M}(T) = (S_{ij})_{n \times n}$ is the discernibility matrix of T , then

$$P_{a_k}(u_i, u_j) = 1 \text{ if and only if } a_k \in S_{ij}.$$

In case of numeric features, instead of the discernibility relation, we will define a fuzzy discernibility relation. If $a_k \in A$ be a real value feature, we define a fuzzy membership function in $U_+ \times U_-$ for the relation $P_{a_k}^{\alpha, \beta}(u_i, u_j)$ as follows:

$$P_{a_k}^{\alpha, \beta}(u_i, u_j) = \begin{cases} 0 & \text{if } d_k(u_i, u_j) \leq \alpha; \\ 1 & \text{if } d_k(u_i, u_j) \geq \beta; \\ \frac{d_k(u_i, u_j) - \alpha}{\beta - \alpha} & \text{otherwise.} \end{cases}$$

where $0 < \alpha < \beta$ are real parameters and $d_k(u_i, u_j) = |a_k(u_i) - a_k(u_j)|$.

Now, we propose the following method for construction of IFPR over the set of features A . For each $a_k \in A$, we define a function $Score_{a_k} : U_- \times U_+ \rightarrow [0, 1]$ such that

$$Score_{a_k}(u_i, u_j) = \frac{1}{\sum_{a_k \in A} P_{a_k}(u_i, u_j)}$$

Intuitively, the value $Score_{a_k}(u_i, u_j)$ determines the probability that a_k should be selected in order to discern u_i from u_j . For any pair of features $a_k, a_l \in A$, we define the following sets:

$$\begin{aligned} X_{kl} &= \{(u_i, u_j) \in U_+ \times U_- : \\ &\quad Score_{a_k}(u_i, u_j) > Score_{a_l}(u_i, u_j)\} \\ Y_{kl} &= \{(u_i, u_j) \in U_+ \times U_- : \\ &\quad Score_{a_k}(u_i, u_j) = Score_{a_l}(u_i, u_j)\} \\ Z_{kl} &= \{(u_i, u_j) \in U_+ \times U_- : \\ &\quad Score_{a_k}(u_i, u_j) < Score_{a_l}(u_i, u_j)\} \end{aligned}$$

Using those sets, we can calculate the following values

$$\begin{aligned} x_{kl} &= \sum_{(u_i, u_j) \in X_{kl}} Score_{a_k}(u_i, u_j) - Score_{a_l}(u_i, u_j) \\ y_{kl} &= \sum_{(u_i, u_j) \in Y_{kl}} Score_{a_k}(u_i, u_j) \\ z_{kl} &= \sum_{(u_i, u_j) \in Z_{kl}} Score_{a_l}(u_i, u_j) - Score_{a_k}(u_i, u_j) \end{aligned}$$

The discernibility IFPR: $P_{dis} = ((\mu_{kl}, \nu_{kl}))_{m \times m}$ as follows

$$\begin{aligned} \mu_{kl} &= \frac{x_{kl}}{x_{kl} + y_{kl} + z_{kl}} \\ \nu_{kl} &= \frac{z_{kl}}{x_{kl} + y_{kl} + z_{kl}} \end{aligned}$$

B. The illustrated examples

Consider an exemplary decision table shown in Table I. This table was created by taking first 10 objects from the famous ‘‘weather data set’’ and adding one more feature (smog) as the fifth feature.

TABLE I: The decision table extended by one new feature.

T_1	a_1	a_2	a_3	a_4	a_5	dec
ID	outlook	temp.	hum.	windy	smog	play
1	sunny	hot	high	FALSE	no	no
2	sunny	hot	high	TRUE	no	no
3	overcast	hot	high	FALSE	yes	yes
4	rainy	mild	high	FALSE	yes	yes
5	rainy	cool	normal	FALSE	no	yes
6	rainy	cool	normal	TRUE	no	no
7	overcast	cool	normal	TRUE	no	yes
8	sunny	mild	high	FALSE	no	no
9	sunny	cool	normal	FALSE	no	yes
10	rainy	mild	normal	FALSE	no	yes

The simplified version \mathbb{M}'_1 of the discernibility matrix for T_1 is presented in Table II. One can see that this new decision table has exactly 2 reducts: $R_1 = \{a_1, a_2, a_4\}$ and $R_2 = \{a_1, a_3, a_4\}$. According to [24], the features a_1 and a_4 are called the *core attributes* and a_5 is the *redundant attribute*.

TABLE II: The simplified discernibility matrix for decision table from Table I.

$\mathbb{M}'(T_1)$	1	2	6	8
3	a_1, a_5	a_1, a_4, a_5	a_1, a_2, a_3, a_4, a_5	a_1, a_2, a_5
4	a_1, a_2, a_5	a_1, a_2, a_4, a_5	a_2, a_3, a_4, a_5	a_1, a_5
5	a_1, a_2, a_3	a_1, a_2, a_3, a_4	a_4	a_1, a_2, a_3
7	a_1, a_2, a_3, a_4	a_1, a_2, a_3	a_1	a_1, a_2, a_3, a_4
9	a_2, a_3	a_2, a_3, a_4	a_1, a_4	a_2, a_3
10	a_1, a_2, a_3	a_1, a_2, a_3, a_4	a_2, a_4	a_1, a_3

Since all features of T_2 are symbolic, the discernibility relations are presented in Table III. The corresponding *Score* functions for features are showing in Table IV.

Now we can calculate the IFPR P_{dis} . Let consider the two features a_1 and a_2 :

$$X_{12} = \{(3, 1), (3, 2), (4, 8), (7, 6), (9, 6), (10, 8)\}$$

$$Z_{12} = \{(4, 6), (9, 1), (9, 2), (9, 8), (10, 6)\}$$

$$Y_{12} = \text{the rest of features.}$$

The sums of scores in previous sets are:

$$x_{12} = 0.5 + 0.33 + 0.5 + 1 + 0.5 + 0.5 = 3.33$$

$$z_{12} = 0.25 + 0.5 + 0.33 + 0.5 + 0.5 = 2.083$$

$$y_{12} = 3.54$$

Therefore:

$$\mu_{12} = \nu_{21} \approx \frac{3.33}{3.33 + 2.083 + 3.54} \approx 0.3759$$

$$\nu_{12} = \mu_{21} \approx \frac{2.083}{3.33 + 2.083 + 3.54} \approx 0.2349$$

The IFPR P_{dis} for decision table T_1 is presented as follows:

$$\begin{pmatrix} (0.50, 0.50) & (0.38, 0.23) & (0.45, 0.19) & (0.51, 0.23) & (0.62, 0.04) \\ (0.23, 0.38) & (0.50, 0.50) & (0.23, 0.08) & (0.41, 0.25) & (0.61, 0.19) \\ (0.19, 0.45) & (0.08, 0.23) & (0.50, 0.50) & (0.39, 0.36) & (0.61, 0.33) \\ (0.23, 0.51) & (0.25, 0.41) & (0.36, 0.39) & (0.50, 0.50) & (0.55, 0.28) \\ (0.04, 0.62) & (0.19, 0.61) & (0.33, 0.61) & (0.28, 0.55) & (0.50, 0.50) \end{pmatrix}$$

TABLE III: The discernibility relation for symbolic features from decision table T_1 .

(u_i, u_j)	P_{a_1}	P_{a_2}	P_{a_3}	P_{a_4}	P_{a_5}
(3,1)	1	0	0	0	1
(3,2)	1	0	0	1	1
(3,6)	1	1	1	1	1
(3,8)	1	1	0	0	1
(4,1)	1	1	0	0	1
(4,2)	1	1	0	1	1
(4,6)	0	1	1	1	1
(4,8)	1	0	0	0	1
(5,1)	1	1	1	0	0
(5,2)	1	1	1	1	0
(5,6)	0	0	0	1	0
(5,8)	1	1	1	0	0
(7,1)	1	1	1	1	0
(7,2)	1	1	1	0	0
(7,6)	1	0	0	0	0
(7,8)	1	1	1	1	0
(9,1)	0	1	1	0	0
(9,2)	0	1	1	1	0
(9,6)	1	0	0	1	0
(9,8)	0	1	1	0	0
(10,1)	1	1	1	0	0
(10,2)	1	1	1	1	0
(10,6)	0	1	0	1	0
(10,8)	1	0	1	0	0

TABLE IV: The *Score* functions for features from T_1 .

(u_i, u_j)	$Score_{a_1}$	$Score_{a_2}$	$Score_{a_3}$	$Score_{a_4}$	$Score_{a_5}$
(3,1)	0.5	0	0	0	0.5
(3,2)	0.33	0	0	0.33	0.33
(3,6)	0.2	0.2	0.2	0.2	0.2
(3,8)	0.33	0.33	0	0	0.33
(4,1)	0.33	0.33	0	0	0.33
(4,2)	0.25	0.25	0	0.25	0.25
(4,6)	0	0.25	0.25	0.25	0.25
(4,8)	0.5	0	0	0	0.5
(5,1)	0.33	0.33	0.33	0	0
(5,2)	0.25	0.25	0.25	0.25	0
(5,6)	0	0	0	1	0
(5,8)	0.33	0.33	0.33	0	0
(7,1)	0.25	0.25	0.25	0.25	0
(7,2)	0.33	0.33	0.33	0	0
(7,6)	1	0	0	0	0
(7,8)	0.25	0.25	0.25	0.25	0
(9,1)	0	0.5	0.5	0	0
(9,2)	0	0.33	0.33	0.33	0
(9,6)	0.5	0	0	0.5	0
(9,8)	0	0.5	0.5	0	0
(10,1)	0.33	0.33	0.33	0	0
(10,2)	0.25	0.25	0.25	0.25	0
(10,6)	0	0.5	0	0.5	0
(10,8)	0.5	0	0.5	0	0

Now we can use the models (A1) and (A2) to find the feature ranking that is additively consistent with P_{dis} . As a result we receive:

$$(w_1, w_2, w_3, w_4, w_5) = (0.468, 0.214, 0.214, 0.104, 0)$$

This means a_1 is the best and a_5 is the worst feature.

Let us consider the decision table T_2 , which is almost the same as T_1 . The only difference is that, a_2 (temperature) and a_3 (humidity) are numeric features.

The discernibility relations for a_1, a_4, a_5 remain unchanged. As an example, for a_2 , we use the fuzzy discernibility relation $P_{a_2}^{\alpha, \beta}$ with $\alpha = 2$ and $\beta = 12$ and for a_3 , we use the

TABLE V: The decision table with numeric features

T_2	a_1	a_2	a_3	a_4	a_5	dec
ID	outlook	temp.(F)	hum.(%)	windy	smog	play
1	sunny	85.0	85.0	FALSE	no	no
2	sunny	80.0	90.0	TRUE	no	no
3	overcast	83.0	86.0	FALSE	yes	yes
4	rainy	70.0	96.0	FALSE	yes	yes
5	rainy	68.0	80.0	FALSE	no	yes
6	rainy	65.0	70.0	TRUE	no	no
7	overcast	64.0	65.0	TRUE	no	yes
8	sunny	72.0	95.0	FALSE	no	no
9	sunny	69.0	70.0	FALSE	no	yes
10	rainy	75.0	80.0	FALSE	no	yes

fuzzy discernibility relation $P_{a_3}^{\alpha, \beta}$ with $\alpha = 5$ and $\beta = 15$. The fuzzy discernibility relations as well as the corresponding *Score* functions for features are presented in Table VI and Table VII.

TABLE VI: The fuzzy discernibility relations for features from decision table T_2 .

(u_i, u_j)	P_{a_1}	P_{a_2}	P_{a_3}	P_{a_4}	P_{a_5}
(3,1)	1	0	0	0	1
(3,2)	1	0.1	0.5	1	1
(3,6)	1	1	1	1	1
(3,8)	1	0.9	0	0	1
(4,1)	1	1	1	0	1
(4,2)	1	0.8	1	1	1
(4,6)	0	0.3	0	1	1
(4,8)	1	0	1	0	1
(5,1)	1	1	1	0	0
(5,2)	1	1	1	1	0
(5,6)	0	0.1	0	1	0
(5,8)	1	0.2	1	0	0
(7,1)	1	1	1	1	0
(7,2)	1	1	1	0	0
(7,6)	1	0	0.4	0	0
(7,8)	1	0.6	1	1	0
(9,1)	0	1	1	0	0
(9,2)	0	0.9	1	1	0
(9,6)	1	0.2	0	1	0
(9,8)	0	0.1	1	0	0
(10,1)	1	0.8	0.8	0	0
(10,2)	1	0.3	1	1	0
(10,6)	0	0.8	0.3	1	0
(10,8)	1	0.1	0	0	0

The *Score* functions can be used to construct the IFPR in the same way as previously. As the result we receive the following matrix:

$$\begin{pmatrix} (0.50, 0.50) & (0.55, 0.22) & (0.36, 0.24) & (0.51, 0.25) & (0.65, 0.06) \\ (0.22, 0.55) & (0.50, 0.50) & (0.16, 0.41) & (0.44, 0.43) & (0.64, 0.28) \\ (0.24, 0.36) & (0.16, 0.41) & (0.50, 0.50) & (0.47, 0.29) & (0.66, 0.20) \\ (0.25, 0.51) & (0.43, 0.44) & (0.29, 0.47) & (0.50, 0.50) & (0.56, 0.25) \\ (0.06, 0.65) & (0.28, 0.64) & (0.20, 0.66) & (0.25, 0.56) & (0.50, 0.50) \end{pmatrix}$$

And the optimal coefficients for this matrix are:

$$(w_1, w_2, w_3, w_4, w_5) = (0.327, 0.227, 0.322, 0.124, 0)$$

We can see that in this case the feature a_3 become almost important as the feature a_1 , and the redundant feature a_5 is still located at the end of the ranking.

TABLE VII: The *Score* functions for features from T_2 .

(u_i, u_j)	$Score_{a_1}$	$Score_{a_2}$	$Score_{a_3}$	$Score_{a_4}$	$Score_{a_5}$
(3,1)	0.5	0	0	0	0.5
(3,2)	0.278	0.028	0.139	0.278	0.278
(3,6)	0.2	0.2	0.2	0.2	0.2
(3,8)	0.345	0.310	0	0	0.345
(4,1)	0.25	0.25	0.25	0	0.25
(4,2)	0.208	0.167	0.208	0.208	0.208
(4,6)	0	0.130	0	0.435	0.435
(4,8)	0.33	0	0.33	0	0.33
(5,1)	0.33	0.33	0.33	0	0
(5,2)	0.25	0.25	0.25	0.25	0
(5,6)	0	0.091	0	0.901	0
(5,8)	0.455	0.091	0.455	0	0
(7,1)	0.25	0.25	0.25	0.25	0
(7,2)	0.33	0.33	0.33	0	0
(7,6)	0.714	0	0.286	0	0
(7,8)	0.278	0.167	0.278	0.278	0
(9,1)	0	0.5	0.5	0	0
(9,2)	0	0.310	0.345	0.345	0
(9,6)	0.455	0.091	0	0.455	0
(9,8)	0	0.091	0.901	0	0
(10,1)	0.385	0.308	0.308	0	0
(10,2)	0.303	0.091	0.303	0.303	0
(10,6)	0	0.381	0.143	0.476	0
(10,8)	0.909	0.091	0	0	0

C. Simplified ranking method.

The presented above method for feature ranking has quite high computational complexity. The time complexity of this proposition is $O((n \cdot m)^2)$, where n is the number of objects and m is the number of attributes. In this Section we propose a heuristic solution called sRAFAR (simplified Rough-fuzzy Algorithm For Attribute Ranking), which is applicable for the data sets with larger number of objects. The idea is to generate a simplified IFPR instead of the full method presented in Section IV.A. In particular, for any continuous feature $a_k \in A$, we discretize its domain into k equal length intervals and use the binary discernibility relation for the discretized feature.

$$A_k = \{(u_i, u_j) \in U_+ \times U_- : u_i, u_j \text{ are discerned by } a_k\}$$

$$= \{(u_i, u_j) \in U_+ \times U_- : P_{a_k}(u_i, u_j) = 1\}$$

Then the simplified IFPR: $P_s = ((\mu'_{kl}, \nu'_{kl}))_{m \times m}$ can be defined by

$$\mu'_{kl} = P(A_k - A_l | A_k \cup A_l) = 1 - \frac{|A_l|}{|A_k \cup A_l|}$$

$$\nu'_{kl} = P(A_l - A_k | A_k \cup A_l) = 1 - \frac{|A_k|}{|A_k \cup A_l|}$$

Thus, μ'_{kl} is the probability that a pair of objects is discernible a_k but not discernible by a_l , and ν'_{kl} is the probability that a pair of objects is discernible a_l but not discernible by a_k . We have the following theorem:

Theorem 1. For any pair of features $a_k, a_l \in A$, the values $|A_k \cup A_l|, |A_k|, |A_l|$ can be calculated in time $O(n)$, where n is the number of objects. Therefore, the heuristic IFPR can be calculated in $O(n \cdot m^2)$, where m is the number of features.

The proof of this fact is similar to the properties of the MD-heuristic in [24].

V. EXPERIMENT RESULTS

In this section, we present the application of our feature ranking methods for the WDBC data set [25]. The WDBC dataset contains features extracted from digitized image of a fine needle aspirate of a breast mass which describes the characteristics of the cell nuclei in the image. This dataset consists of 569 instances with 30 attributes and two decision classes. The features are encoded by V_1, V_2, \dots, V_{30} . We will compare the quality of feature ranking lists generated by:

- RAFAR: Rough-fuzzy Algorithm For Attribute Ranking;
- sRAFAR: simplified version of RAFAR;
- Random Forest Feature Importance; ¹
- No ranking, i.e. using the original feature list: V_1, V_2, \dots, V_{30} .

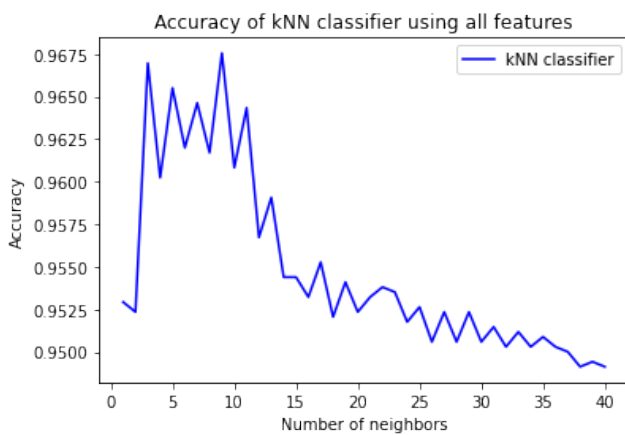


Fig. 2: The accuracy of kNN classifier for different values of k . The highest accuracy is achieved for $k = 9$.

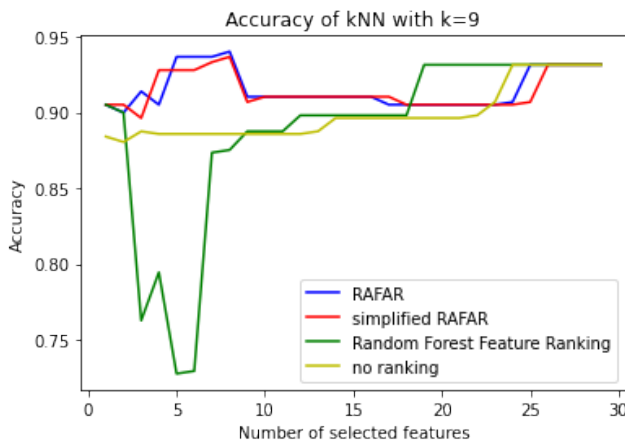


Fig. 3: Comparison of ranking lists with respect to kNN classifier with $k = 9$.

In order to analyze the quality of a ranking list of features (attributes), we select a classifier (classification algorithm) and apply it to the sub-dataset restricted to the first m features

¹https://scikit-learn.org/stable/auto_examples/inspection/plot_permutation_importance_multicollinear.html

for $m = 1, 2, \dots, 30$. For a fixed value of m we evaluate the accuracy of the classifier using 5-fold-cross-validation technique.

The first classifier in our experiment is kNN. Figure 2 presents the accuracy of kNN on the whole data set for different values of the parameter k . We can see that the optimal value of k for kNN classifier equals 9. Therefore we will select kNN with $k = 9$ in the first experiment.

In Fig. 3 the accuracy of kNN with $k = 9$ using first m features of the ranking list for $m = 1, 2, \dots, 30$ is presented. One can see that the accuracy of ranking lists generated by both of our algorithms outperform the other ranking lists.

In Fig. 4 the accuracy comparison of decision tree classifier using first features in the ranking lists is presented. We can see that in this case, the ranking list generated by the RAFAR algorithm seems to be best, especially when we want to use up to 17 features.

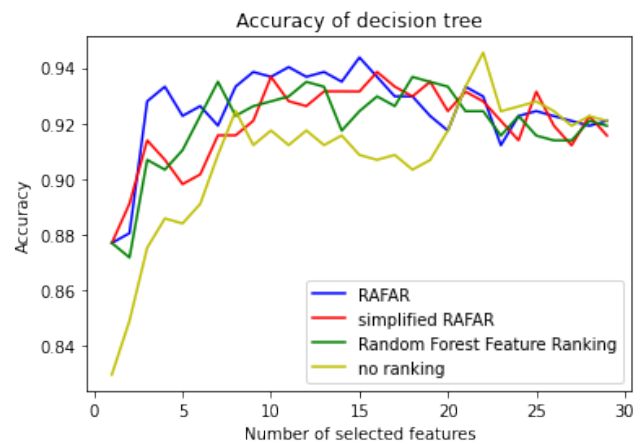


Fig. 4: Comparison of ranking lists with respect to decision tree.

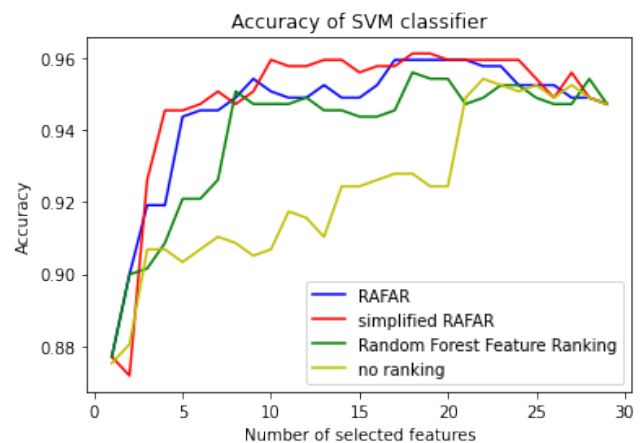


Fig. 5: Comparison of ranking lists with respect to SVM classifier.

Figures 5 present the accuracy comparison for SVM classifier. In this case we notice the fact that the ranking list generated by the sRAFAR algorithm is the best one.

VI. CONCLUSIONS AND FUTURE RESEARCH

In this paper we proposed a new method for feature ranking. We constructed the Intuitionistic Fuzzy Preference Relation (IFPR) for the set of features and searched for the optimal feature ranking that is consistent with the IFPR. All experiments are showing that the proposed rough-fuzzy algorithms for attribute ranking outperform the state of the art method (Random Forest Feature Ranking is the main feature ranking method in the Python scikit-learn library <https://scikit-learn.org/stable/>). We can conclude that the proposed methods are promising and should be thoroughly investigated.

One of the future research direction is multi-criteria feature ranking instead of a single preference relation defined on the based of discernibility power of the features as t has been proposed in RAFAR. The general framework is shown in Fig 1. This idea is motivated by the real life decision making process, where a decision is usually made by a group of experts, $E_k (k = 1, 2, \dots, m)$ with different weights $\lambda = (\lambda_1, \dots, \lambda_m)$, where $\sum_{k=1}^m \lambda_k = 1$ and $\lambda_k \geq 0$ for $k = 1, \dots, k$. In such cases, the individual preference relations of the experts are aggregated to derive a collective preference relation. Let $B^{(k)} = ((\mu_{ij}^{(k)}, \nu_{ij}^{(k)}))_{n \times n}$ be the intuitionistic fuzzy preference relation of the expert E_k , the aggregated preference relation \bar{B} is defined by $\bar{B} = \sum_{k=1}^m \lambda_k \cdot B^{(k)}$. In other words $\bar{B} = ((\bar{\mu}_{ij}, \bar{\nu}_{ij}))_{n \times n}$, where

$$\bar{\mu}_{ij} = \sum_{k=1}^m \lambda_k \mu_{ij}^{(k)}; \quad \bar{\nu}_{ij} = \sum_{k=1}^m \lambda_k \nu_{ij}^{(k)};$$

Theorem 2. *If $B^{(k)}$ are intuitionistic fuzzy preference relation of the expert E_k for $k = 1, \dots, m$ and the weight vector $\lambda = (\lambda_1, \dots, \lambda_m)$ is a probabilistics vector, i.e. $\sum_{k=1}^m \lambda_k = 1$ and $\lambda_k \geq 0$ for $k = 1, \dots, k$, then \bar{B} is also an intuitionistic fuzzy preference relation.*

In such situations, we can apply both ranking methods (i.e. either the models (A1), and (A2) for the additive consistency requirement or the models (M1) and (M2) for the multiplicative consistency requirement) for the collective intuitionistic fuzzy preference relation \bar{B} .

Following this idea, in case of feature ranking problem, we can create more IFPR with different aspects and include them into the calculation process. For example, another preference relation could be calculate on the base of the class homogeneity of features.

We also plan to verify the accuracy of RAFAR and sRAFAR for bigger and more challenging data sets.

ACKNOWLEDGMENT

The authors would like to thank the Vietnam Institute for Advanced Study in Mathematics (VIASM) for the support during the time they had been visiting and working at the Institute.

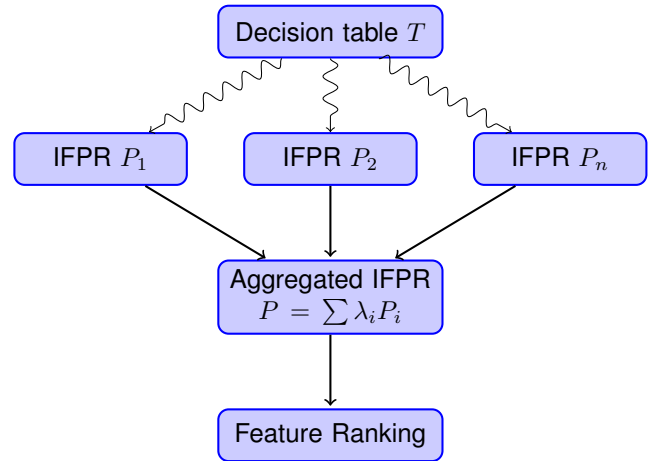


Fig. 6: The general framework for multi-criteria feature ranking.

REFERENCES

- [1] S. K. Pal and A. Skowron, *Rough-Fuzzy Hybridization: A New Trend in Decision Making*, 1st ed. Berlin, Heidelberg: Springer-Verlag, 1999.
- [2] J. Buckley, "Fuzzy hierarchical analysis," *Fuzzy Sets and Systems*, vol. 17, no. 3, pp. 233–247, 1985. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/0165011485900909>
- [3] Y. Dong, Y. Xu, and H. Li, "On consistency measures of linguistic preference relations," *European Journal of Operational Research*, vol. 189, no. 2, pp. 430–444, 2008. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0377221707005619>
- [4] S. Orlovsky, "Decision-making with a fuzzy preference relation," *Fuzzy Sets and Systems*, vol. 1, no. 3, pp. 155–167, 1978. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/0165011478900015>
- [5] J. Tang, F. Meng, and Y. Zhang, "Decision making with interval-valued intuitionistic fuzzy preference relations based on additive consistency analysis," *Information Sciences*, vol. 467, pp. 115–134, 2018. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0020025518305565>
- [6] Z.-J. Wang and X. Tong, "Consistency analysis and group decision making based on triangular fuzzy additive reciprocal preference relations," *Information Sciences*, vol. 361–362, pp. 29–47, 2016. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0020025516303000>
- [7] V. Traneva, S. Tranev, and D. Mavrov, "Interval-valued intuitionistic fuzzy decision-making method using index matrices and application in outsourcing," in *Proceedings of the 16th Conference on Computer Science and Intelligence Systems, Online, September 2-5, 2021*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, M. Paprzycki, and D. Slezak, Eds., vol. 25, 2021, pp. 251–254. [Online]. Available: <https://doi.org/10.15439/2021F77>
- [8] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, pp. 338–353, 1965. [Online]. Available: <http://www-bisc.cs.berkeley.edu/Zadeh-1965.pdf>
- [9] K. T. Atanassov, "Intuitionistic fuzzy sets," *Fuzzy Sets Syst.*, vol. 20, no. 1, p. 87–96, aug 1986.
- [10] H. Bustince and P. J. Burillo, "Structures on intuitionistic fuzzy relations," *Fuzzy Sets Syst.*, vol. 78, no. 3, pp. 293–303, 1996. [Online]. Available: [https://doi.org/10.1016/0165-0114\(96\)84610-0](https://doi.org/10.1016/0165-0114(96)84610-0)
- [11] M. Pal, S. K. Khan, and A. K. Shyamal, "Intuitionistic fuzzy matrices," *Notes on Intuitionistic fuzzy sets*, vol. 8, no. 2, pp. 51–62, 2002.
- [12] K. T. Atanassov, *Intuitionistic Fuzzy Relations (IFRs)*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 147–193. [Online]. Available: https://doi.org/10.1007/978-3-642-29127-2_8
- [13] Z. Pawlak, "Rough sets," *Int. J. Parallel Program.*, vol. 11, no. 5, pp. 341–356, 1982. [Online]. Available: <https://doi.org/10.1007/BF01001956>

- [14] A. Skowron and C. Rauszer, "The discernibility matrices and functions in information systems," in *Intelligent Decision Support - Handbook of Applications and Advances of the Rough Sets Theory*, ser. Theory and Decision Library, R. Slowinski, Ed. Springer, 1992, vol. 11, pp. 331–362. [Online]. Available: https://doi.org/10.1007/978-94-015-7975-9_21
- [15] Z. Pawlak and A. Skowron, "Rudiments of rough sets," *Information Sciences*, vol. 177, no. 1, pp. 3–27, January 2007.
- [16] D. Slezak, "Rough sets and few-objects-many-attributes problem: The case study of analysis of gene expression data sets," in *Frontiers in the Convergence of Bioscience and Information Technologies 2007, FBIT 2007, Jeju Island, Korea, October 11-13, 2007*, D. Howard and P. Rhee, Eds. IEEE Computer Society, 2007, pp. 437–442. [Online]. Available: <https://doi.org/10.1109/FBIT.2007.160>
- [17] H. S. Nguyen, *Approximate Boolean Reasoning: Foundations and Applications in Data Mining*. Berlin, Heidelberg: Springer-Verlag, 2006, p. 334–506.
- [18] X. Jia, L. Shang, B. Zhou, and Y. Yao, "Generalized attribute reduct in rough set theory," *Knowl. Based Syst.*, vol. 91, pp. 204–218, 2016. [Online]. Available: <https://doi.org/10.1016/j.knosys.2015.05.017>
- [19] D. Slezak and J. Wroblewski, "Order based genetic algorithms for the search of approximate entropy reducts," in *Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing, 9th International Conference, RSFDGrC 2003, Chongqing, China, May 26-29, 2003, Proceedings*, ser. Lecture Notes in Computer Science, G. Wang, Q. Liu, Y. Yao, and A. Skowron, Eds., vol. 2639. Springer, 2003, pp. 308–311. [Online]. Available: https://doi.org/10.1007/3-540-39205-X_45
- [20] Z. Xu and H. Liao, "A survey of approaches to decision making with intuitionistic fuzzy preference relations," *Know.-Based Syst.*, vol. 80, no. C, p. 131–142, may 2015. [Online]. Available: <https://doi.org/10.1016/j.knosys.2014.12.034>
- [21] H. Torun, "Group decision making with intuitionistic fuzzy preference relations," *Knowledge-Based Systems*, vol. 70, 04 2014.
- [22] P. Ren, Z. Xu, and J. Kacprzyk, "Group decisions with intuitionistic fuzzy sets," *Handbook of Group Decision and Negotiation*, pp. 977–995, 2021.
- [23] Z. Xu and R. R. Yager, "Intuitionistic and interval-valued intuitionistic fuzzy preference relations and their measures of similarity for the evaluation of agreement within a group," *Fuzzy Optimization and Decision Making*, vol. 8, pp. 123–139, 2009.
- [24] L. G. Nguyen and H. S. Nguyen, "On elimination of redundant attributes from decision table," in *Federated Conference on Computer Science and Information Systems - FedCSIS 2012, Wroclaw, Poland, 9-12 September 2012, Proceedings*, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., 2012, pp. 317–322. [Online]. Available: <https://fedcsis.org/proceedings/2012/pliki/324.pdf>
- [25] W. N. Street, W. H. Wolberg, and O. L. Mangasarian, "Nuclear feature extraction for breast tumor diagnosis," in *Biomedical Image Processing and Biomedical Visualization*, R. S. Acharya and D. B. Goldgof, Eds., vol. 1905, International Society for Optics and Photonics. SPIE, 1993, pp. 861 – 870. [Online]. Available: <https://doi.org/10.1117/12.148698>

15th International Workshop on Computational Optimization

MANY real world problems arising in engineering, economics, medicine and other domains can be formulated as optimization tasks. These problems are frequently characterized by non-convex, non-differentiable, discontinuous, noisy or dynamic objective functions and constraints which ask for adequate computational methods.

The aim of this workshop is to stimulate the communication between researchers working on different fields of optimization and practitioners who need reliable and efficient computational optimization methods.

TOPICS

The list of topics includes, but is not limited to:

- combinatorial and continuous global optimization
- unconstrained and constrained optimization
- multiobjective and robust optimization
- optimization in dynamic and/or noisy environments
- optimization on graphs
- large-scale optimization, in parallel and distributed computational environments
- meta-heuristics for optimization, nature-inspired approaches and any other derivative-free methods
- exact/heuristic hybrid methods, involving natural computing techniques and other global and local optimization methods
- numerical and heuristic methods for modeling

The applications of interest are included in the list below, but are not limited to:

- classical operational research problems (knapsack, traveling salesman, etc)
- computational biology and distance geometry
- data mining and knowledge discovery
- human motion simulations; crowd simulations
- industrial applications
- optimization in statistics, econometrics, finance, physics, chemistry, biology, medicine, and engineering
- environment modeling and optimization

BEST PAPER AWARD

The best WCO'22 paper will be awarded during the social dinner of FedCSIS 2022.

The best paper will be selected by WCO'22 co-Chairs by taking into consideration the scores suggested by the reviewers, as well as the quality of the given oral presentation.

TECHNICAL SESSION CHAIRS

- **Fidanova, Stefka**, Bulgarian Academy of Sciences, Bulgaria
- **Mucherino, Antonio**, INRIA, France
- **Zaharie, Daniela**, West University of Timisoara, Romania

GaMeDE2 — improved Gap-based Memetic Differential Evolution applied to Multimodal Optimization

Michał Antkiewicz, Paweł B. Myszowski
Wrocław University of Science and Technology
Faculty of Information and Communication Technology
ul. Ignacego Łukasiewicza 5, 50-371 Wrocław, Poland
email: {michal.antkiewicz, pawel.myszowski}@pwr.edu.pl

Maciej Laszczyk
email: maciej.laszczyk@gmail.com

Abstract—This paper presents an improved Gap-based Memetic Differential Evolution (GaMeDE2), the modification of the GaMeDE method, which took second place in the GECCO 2020 Competition on Niching Methods for Multimodal Optimization. GaMeDE2 has reduced complexity, fewer parameters, redefined initialization, selection operator, and removed processing phases. The method is verified using standard benchmark function sets (classic ones and CEC2013) and a newly proposed benchmark set comprised of deceptive functions. A detailed comparison to state-of-the-art methods (like HVCMO and SDLCSDE) is presented, where the proposed GaMeDE2 outperforms or gives similar results to other methods. The document is concluded by discussing various insights on the problem instances and the methods created as a part of the research.

I. INTRODUCTION

MULTIMODAL Optimization (MMO) is a well-established problem in the literature (e.g.[1]). Due to its practical applications, it has been studied over the years. In real-world practical problems, the number of optimal solutions is not known *a priori*. It means, that valuable results might be ruled out by the assumption of sole optimal solution existence. As a result of real-life unpredictability and constraints, the optimal solution might not be feasible, and a suboptimal one may be preferred. What is more, the knowledge of other valuable solutions can support decision-making and allows for agile switches without the need for running the optimization process from the beginning with the risk of convergence to the same optima, which is the drawback of the standard unimodal optimization approaches.

The well-established competition in the multimodal optimization area is the annual GECCO Conference and Competition on Niching Methods for Multimodal Optimization based on the benchmark function suit [2]. GaMeDE2 proposed in this article is an extension of the GaMeDE [3] method, which took second place in the 2020 competition edition. The paper presents several modifications to boost the generality, which has been shown on the additional benchmark suits. Three other methods were selected to compare the final results. Self-adaptive Double-Layer-Clustering Speciation Differential

Evolution (SDLCSDE) [4] is the recently published approach and gives very promising results. Unfortunately, it has not been evaluated on the GECCO Competition benchmark set, and the source code was not available to carry out the experiments - the results for another benchmark set of classical multimodal functions [5] have been used. The next reference method used is the Hill-Valley-Clustering-based VMO (HVCMO) [6], a novel method based on the HillVallea [7] - a winning method in GECCO 2019 Competition on Niching Methods for Multimodal Optimization. The third compared method is the original GaMeDE approach.

The rest of the article is structured as follows. Section 2 covers the short literature review related to multimodal problems. The proposed GaMeDE2 method is introduced in section 3. The experimental setup, datasets descriptions, and results for the evaluated methods with the discussion of the results are given in Section 4. Lastly, the paper is concluded in section 5.

II. SCIENTIFIC BACKGROUND AND RELATED WORKS

A series of articles show the importance of multimodal problems in multiple practical and various valuable areas, such as Drug Scheduling Problem, Protein Structure Prediction, Resource-Constrained Multiproject Scheduling Problem, cosmological applications, and an iconic machine-learning classification problem (e.g [8]) and many others. In literature, multimodality tends to be combined with multi-objectiveness, wherefore research in one area benefits both.

Besides real-world applications, sets of benchmark functions for the MMO have been proposed over the years. They are either manually fabricated to emphasise certain features or forged by combining multiple unimodal benchmark functions. Among the most recognizable in the literature [9] are multimodal benchmark functions based on: Rastrigin's function, Shubert's function, Vincent's function, Griewank's function, Schwefel's function, and Ackley's function. As a result of the number of combinations, a unified evaluation set has been introduced [2]. It consists of highly varied functions in terms of number of dimensions (1-20), domain and peak height scale, number of local and global optima (0-many/1-216),

optima distribution, and landscape variability. Additionally, it contains a single instance of a deceptive function - the Five-Uneven-Peak Trap. This set has been used in the GECCO 2020 Niching Competition on Multimodal Optimization and the following editions. The alternative function composition has been proposed in [10] which shows the flexibility of combining the functions. Another benchmark set grounded in the literature [5] introduces new instances related to the competition set and modifies the budget for the subset of those already present. These functions are divided into six categories: Deceptive Functions, 1D Multimodal Functions, 2D Multimodal Functions, ≥ 2 D More Challenging Multimodal Functions, Inverted Rastrigin Function, and Generic Hump Functions. The added deceptive/trap functions are: 1D Two-Peak Trap and 1D Central Two-Peak Trap. The deceptiveness factor is an essential aspect of Multimodal Optimization [11]. There are limited works related to high dimensional multimodal trap functions. The one proposed approach is to apply function composition [12], yet it has not been openly adapted as the benchmark function, nor extended research in the area has been found at the moment of writing this article.

As stated in the introduction, a diverse set of valuable solutions is expected when solving a multimodal problem. For this reason, an efficient exploratory method is crucial. Evolutionary Computation is known to be effective in complex multimodal optimization, e.g. [1][13][12][14]. A significant amount of attention has been paid to Continuous Multimodal Optimization, where Differential Evolution (DE) [15] is a reference method. It is still a widely used approach in the literature. Work [16] introduces a Dual-Strategy Mutation, adaptive individual selection, and converged individuals archive. Authors in [17] proposed a novel mutation strategy based on the Local Binary Pattern used for niche detection. Another modification was introduced in [18], where Distributed Individual for Multiple Peaks (DIMP) has been used to track optima. It has been extended by adding two novel mechanisms. First, Lifetime Mechanism is inspired by the natural phenomenon of organism aging and limited lifespan. Second, Elite Learning Mechanism (ELM) is introduced to refine the accuracy and efficiency of the archiving mechanism.

High-quality solutions in the multimodal landscape may often be found in different parts of the search space, complicating the single population convergence. Niching's [8] technique was introduced to prevent this effect by dividing the domain into multiple subsets called niches. The general idea in multimodal optimization is to detect niches, ideally located around the optima, and explore them separately. Niching is a widely applied concept in MMO and several modifications can be found in the literature. Nearest-neighbor niching introduced in [19] aids in achieving a well-balanced species. Another approach to increasing the solutions' (swarm) diversity is introducing the Equilibrium Factor to modify the individual's velocity [20]. Parameter-less niching mechanism (affinity propagation clustering) [21] is a valuable direction that reduces the method's parameter number and helps to locate the nearest peak, which boosts the convergence. Double-layer-clustering

[4] has been proposed to increase the diversity and global exploration to locate more global optima. The DE method is applied on the niche level to support escaping local optima and detecting new promising areas in the search space. It also benefits from a self-adaptive strategy to reduce the number of parameters by self-adapting the crossover probability and scaling factor used by DE. On the other hand, it still requires the population size defined per problem instance. Hill-ValLEA [22] is the GECCO 2019 Niching Competition on Multimodal Optimization competition winner. It introduced the Hill-Valley Clustering approach to adaptively cluster the search space in niches residing around a single optimum. It utilizes the Hill-Valley test [23] to determine whether two solutions belong to the same niche. Combined with a core search algorithm to optimize the niches and restart scheme, it outperformed its competitors. Hill-Valley-Clustering-based VMO (HVcMO) [6] merges the HillValLEA method with Variable Mesh Optimization [24], which significantly boosted the optima detection and improved the efficient use of budget.

The original GaMeDE [3] is a novel method, drawing concepts from the MMO using DE as its base. It benefits from GAP selection (and archive management mechanism) to keep high diversity, clustering for identification of promising areas, and local search optimization. Its core functionality is a two-phase approach - the *WIDE* phase uses standard random selection followed by the HillValley Clustering to split the population into niches. Each niche is further optimized by the AMaLGaM Univariate[25] local optimizer, and the best individual per cluster is stored in the archive. *FOCUS* phase uses two selections alternately - standard tournament selection and GAP selection. The latter is a novel approach proposed by the authors. It follows the idea of tournament selection, but instead of fitness (or objective function value), it uses a 'gap' distance. The 'gap' distance is simply the Euclidean distance to the nearest existing individual in the decision space. Its goal is to guide the exploration of 'blank spaces' of the landscape. The last step of the *FOCUS* phase is the HillClimber local optimization of new points in the archive. The method starts with the *FOCUS* phase and switches to the *WIDE* when no longer can find new optima.

III. PROPOSED METHOD

GaMeDE2 is a redesigned GaMeDE [3] method developed for the GECCO 2020 Competition on Niching Methods for MMO. It means a high probability of overtuning and overfitting for the competition benchmark suite. Experiments conducted using new instances presented a lack of generalization and exposed the need for improvements in this area. Mentioned results are shown in the experiments chapter. The goal of modifications introduced in this work is to simplify the original GaMeDE algorithm and improve its effectiveness across multiple benchmark sets. The main objective of the MMO aspects remains unchanged. In this article, it is defined as the search for global optima only (in contrast to the search for all, including local, optima), which is a common approach in literature [3], [4], [25]. Each multimodal problem instance

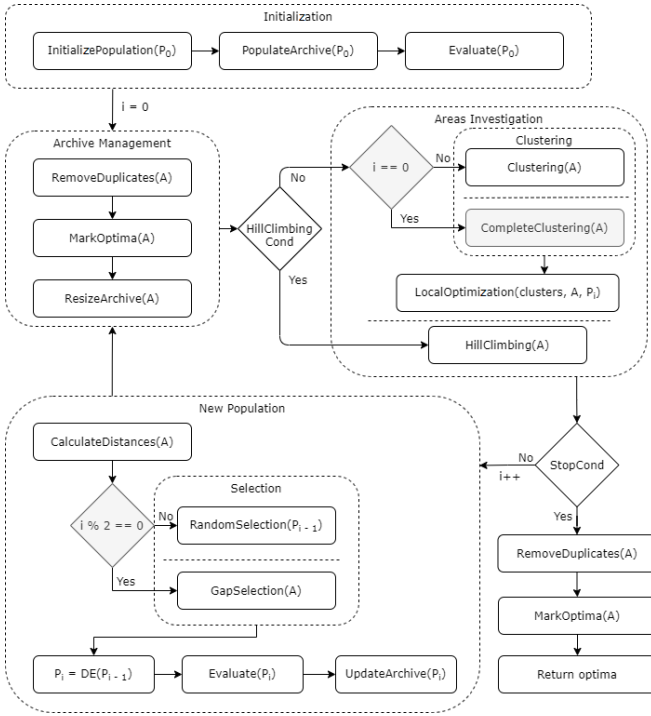


Fig. 1. A general schema of GaMeDE2

is being solved independently with an objective function described as:

$$\max_x f(x) = y, \quad x \in R^d \quad (1)$$

The aim is to find as many different x vectors as possible where the minimum distance between two optima is given.

In this section, GaMeDE2 with a set of introduced modifications has been described. The Algorithm 1 presents the complete version of the GaMeDE2 pseudo-code, while Figure 1 illustrates the modifications described below.

GaMeDE2 is based mainly on a DE that explores/exploits new areas in the problem landscape. DE is steered by the archive (concept strictly incorporated from multi-objective problems), which stores candidates for global optima, and concentrates on 'gaps' between already found solutions. DE optimization is additionally boosted by a clustering mechanism to identify promising areas. Subsequently, local search speeds up the convergence in those areas. The *HillClimbing* procedure further optimizes local optima.

GaMeDE2 initialization (see The Algorithm 1 or/and Figure 1) starts with a random initialization as presented in pseudocode line 3, the initial population is evaluated and used to populate an archive. The archive is used to store the global optima and promising individuals. The *PopulateArchive* method tries to add each individual to the archive. It utilizes the simplified HillValley Test[7] - creates the middle-point M between the tested individual A and his nearest neighbor B in the archive. If the middle-point is worse than both points (A and B), they are kept as there is a chance they come from two

different niches. Otherwise, only the best individual (A , B , or C) is stored. In lines 6 and 12, distances between points in the archive are calculated and stored for future usage. In lines 7 and 26, *ArchiveManagement* takes place. Its pseudocode is presented in Algorithm 2 and remained unchanged from the original GaMeDE version. It removes duplicates from the archive, marks current optima, and truncates the archive by sorting it and removing worst solutions until it fits the *MaxArchiveSize*.

Algorithm 1 GaMeDE2 pseudocode

```

1: PrevGenOptima,  $i \leftarrow 0$ 
2: Optima  $\leftarrow \emptyset$ 
3:  $P_i \leftarrow \text{InitRandomPop}()$ 
4: Evaluate( $P_i$ )
5:  $A \leftarrow \text{PopulateArchive}(P_i)$ 
6: CalculateDistances( $A$ )
7: ArchiveManagement( $A$ , Optima)
8: Clusters  $\leftarrow \text{CompleteClustering}(A)$ 
9: LocalOptimization(Clusters,  $A$ ,  $P_i$ )
10: while !StopCondition() do
11:    $i++$ ;
12:   CalculateDistances( $A$ )
13:    $P_i \leftarrow \emptyset$ 
14:   while  $|P_i| \neq |P_{i-1}|$  do
15:     if  $i \% 2 == 0$  then
16:       Parents  $\leftarrow \text{GapSelection}(A)$ 
17:     else
18:       Parents  $\leftarrow \text{RandomSelection}(P_{i-1})$ 
19:     end if
20:     Mutants  $\leftarrow \text{Mutate}(\text{Parents})$ 
21:     Children  $\leftarrow \text{Crossover}(\text{Mutants})$ 
22:      $P_i \leftarrow P_i + \text{Children}$ 
23:     Evaluate( $P_i$ )
24:      $A \leftarrow \text{UpdateArchive}(P_i, A)$ 
25:   end while
26:   ArchiveManagement( $A$ , Optima)
27:   if  $|Optima| - \text{PrevGenOptima} > \text{MinNewOptima}$  then
28:     HillClimbing( $A$ )
29:   else
30:     Clusters  $\leftarrow \text{Clustering}(A)$ 
31:     LocalOptimization(Clusters,  $A$ ,  $P_i$ )
32:   end if
33:   PrevGenOptima  $\leftarrow |Optima|$ 
34: end while
35:  $A \leftarrow \text{RemoveDuplicates}(A)$ 
36: Optima  $\leftarrow \text{MarkOptima}(A)$ 
37: Return: Optima

```

Before the first loop starts, *CompleteClustering* takes place (see line 8). It applies for an additional clustering pass as described in the modifications section. In line 9, *LocalOptimization* utilizes AMaLGaM Univariate (as in the original GaMeDE). Lines 10-34 present the main loop, which runs until *StopCondition* is fulfilled - in this case, until the budget (number of evaluated individuals) is fully used. A new, empty population is initialized in line 13 and populated in lines 14-25. It uses *GapSelection* and *RandomSelection* alternately without the need for having any phases as described in the modifications section.

Lines 20-23 present a standard DE process. New individuals are processed in twos. Each of the two individuals returned

from the selection undergoes the basic DE mutation process (based on two different, randomly selected, individuals and factor F). Genes are truncated to make sure all individuals are feasible. In the end, Uniform Crossover is applied. In line 24, the archive is updated with the same procedure used in *PopulateArchive*. Lines 27-32 present the local optimization step, which takes place at the end of each algorithm iteration. The default path is to run *LocalOptimization* for the promising clusters (see line 9). Alternative *HillClimbing* method is used if many new optima candidates appear in this iteration.

Algorithm 2 ArchiveManagement pseudocode

```

1: Params:  $A, Optima$ 
2:  $A \leftarrow RemoveDuplicates(A)$ 
3:  $Optima \leftarrow MarkOptima(A)$ 
4:  $A \leftarrow ResizeArchive(MaxArchiveSize, Optima, A)$ 

```

GaMeDE2 method is based on the original GaMeDE, however it comprises several modifications that reduce the complexity of the method.

A. GaMeDE2 – major proposed modifications

GaMeDE2 consists of several modifications as follows:

- 1) **Skip first selection** - In the GaMeDE, the first selection took place right after the population initialization. Both selection and initialization in GaMeDE are random, which does not introduce any improvement other than increasing the initial pool of random search. However, the evaluation cost is doubled, which might significantly reduce the number of further evaluations for the low-budget problem instances.
- 2) **Remove phase switching** - Based on the conducted experiments, it is not necessary to switch between *WIDE* and *FOCUS* phases between generations. Local optimization performed by a *HillClimber* stays crucial for a group of problems, but it does not have to be paired with global algorithm phases. The condition to run a *HillClimber* in Area investigation step remains unchanged, but it is not propagated further to the next steps.
- 3) **Double initial clustering** - An alternative clustering has been proposed to be used in the initial generation. Its purpose is to detect all the 'easy' optima faster. In the base algorithm, points found in the archive are spread around the search space, and many lay in the same niche. While it might increase the diversity in population, it significantly increases the number of clusters to search and the chance of crossing over the given budget. In GaMeDE2, an alternative approach to cluster generation uses candidates from the archive focusing on new areas. Solutions are clustered using additional Hill-Valley Clustering described in [7] to further reduce the number of candidates from the same global optimum (as an attractor). The Hill-Valley Test itself is simplified by reducing the middle points count to one. Adding second clustering was inspired by the approach in [4], where

performing another DE iteration, based on top of the seeds found in the first pass, appeared to be successful. However, in this work, only the clustering was repeated with no additional DE run.

- 4) **Selection type is not related to phase** - Results of experiments confirm that the selection phases do not have to be bound to the *WIDE/FOCUS* phases. However, the experiments showed that it is still crucial to keep both *Random* and *GAP* selections. Those two selections are used alternately through the subsequent generations, which allows to fully drop the need for defining two phases. The experiments showed that such selection composition gives the best results: one (*Random*) provides high diversity, while the other (*GAP*) focuses on search in poorly explored areas.

Both methods GaMeDE and GaMeDE2 are verified using 3 benchmark datasets, and the results are compared to two state-of-the-art methods. The research results are presented in the next section.

IV. EXPERIMENTS

Modifications proposed in GaMeDE2 have been experimentally verified across three different test sets. Each problem instance has been evaluated, and results are compared using GaMeDE, GaMeDE2 and Hill-Valley-Clustering-based VMO (HVcMO) [6]. For the second test set, results have also been compared with the recently presented Double-layer-clustering (SDLCSDE) [4]. Unfortunately, SDLCSDE cannot be used as the reference method for all test sets – the source code was not available to perform the experiments.

A. Setup

MMO aims to find as many global optima as possible in a budget defined per each problem instance. The only metric used is the **Peak Ratio** (PR) which is a fraction of the global optima detected. Thus, the **Success Rate** (SR) has been calculated as the number of runs with all optima detected divided by the number of all runs. To verify if the global optimum has been reached, accuracy $\epsilon = 10^{-5}$ has been selected, the same as in [7]. For the SDLCSDE, accuracy levels were different across the test set. To compare performance precisely, GaMeDE2 has been tested on the problem instances where SDLCSDE accuracy levels were higher than the standard.

Both methods (GaMeDE and GaMeDE2) include non-deterministic elements, and experiments were repeated 30 times on all problem instances to average the results. The Wilcoxon signed-rank test has been applied to verify statistical significance using averaged results. The key advantage of the GaMeDE and GaMeDE2 is their generality, which means they can be successfully applied to a set of different problem instances without any configuration changes. Therefore, for both methods, only a single configuration has been used – in contrast to SDLCSDE, where the 'Population size' parameter was manually selected per each instance – which is not efficient while solving new, unknown problem instances.

To tune GaMeDE2 and find its optimal configuration, 5-Level Taguchi [26] Parameter Design procedure has been used. A set of experiment configurations was generated using an orthogonal matrix, and each configuration was repeated 10 times. This procedure was further repeated for a subset of test functions. The parameters with the highest *Signal-to-Noise* change were fine-tuned first based on the average results. All the parameters have been processed in that manner, subsequently in the descending order of Signal-to-Noise change. Table I presents selected values used by GaMeDE2. The GaMeDE uses the configuration proposed in [3].

TABLE I
GAMEDE2 – BEST FOUND CONFIGURATION

Parameter	Value
PopulationSize	1000 * dim
TournamentSize	10
MaxArchiveSize	25 * dim^2
DiversityPhaseMinNewOptima	5
LocalOptInitialStep	0.01
MutationProbability	0.6
CrossoverProbability	0.2
F	0.01

All presented experiments were carried out on three test sets, consisted of multimodal real-value problems.

B. Test sets

The key attributes of each instance are presented in tables (see. Tab.II, Tab.III, Tab.IV and Tab.X) – it shows the number of global and local optima, number of dimensions and fitness evaluation budget. Three test sets are used in research: **CEC2013**, **Classical** Functions and **Deceptive** Functions, presented respectively in the rest of this section.

First test set - **CEC2013** / **GECCO2020** presented in Table II is commonly used in literature benchmark (e.g. [6][3]) and the same set which has been used in GECCO 2020 Competition on Niching Methods for Multimodal Optimization. It contains a variety of functions with different properties such as: deceptiveness (F1), wide spread of global optima count (F3 vs F9), wide spread of local optima count (F4 vs F6), composite functions (CF11 - CF20), flat and steep niches (CF20).

The second test set, called **Classical** Functions presented in Table III is a benchmark described in [5] and the one that has been selected by the authors of SDLCSDE [4]. The test set contains a number of functions already introduced in the CEC2013 set, yet it significantly decreases the budget given for each instance. The only fully repeated entries are B13 and B14. It also introduces a few new variants: Two-Peak Trap, Central Two-Peak Trap, Decreasing Maxima, Uneven Maxima, and Shekel's Foxholes. Due to the small budget, this set is used to verify the efficiency of the algorithms.

The last test set (**Deceptive** Functions, see Table IV) has been proposed to explore further the aspect of resistance to deceptive traps, which is an essential aspect in the optimization area. It is based on the Classical Functions set, and it consists

TABLE II
CEC2013 / GECCO2020 MULTIMODAL FUNCTION SET

#	Function Name	D	#GOPT	#LOPT	Budget
F1	Five-Uneven-Peak Trap	1	2	3	50K
F2	Equal Maxima	1	5	0	50K
F3	Uneven Decreasing Maxima	1	1	4	50K
F4	Himmelblau	2	4	0	50K
F5	Six-Hump Camel Back	2	2	5	50K
F6	Shubert	2	18	many	200K
F7	Vincent	2	36	0	200K
F8	Shubert	3	81	many	400K
F9	Vincent	3	216	0	400K
F10	Modifier Rastrigin	2	12	0	200K
CF11	Composite Function 1	2	6	many	200K
CF12	Composite Function 2	2	8	many	200K
CF13	Composite Function 3	2	6	many	200K
CF14	Composite Function 3	3	6	many	400K
CF15	Composite Function 4	3	8	many	400K
CF16	Composite Function 3	5	6	many	400K
CF17	Composite Function 4	5	8	many	400K
CF18	Composite Function 3	10	6	many	400K
CF19	Composite Function 4	10	8	many	400K
CF20	Composite Function 4	20	8	many	400K

TABLE III
CLASSICAL MULTIMODAL BENCHMARK FUNCTION SET

#	Function Name	D	#GOPT	#LOPT	Budget
B1	Two-Peak Trap	1	1	1	10K
B2	Central Two-Peak Trap	1	1	1	10K
B3	Five-Uneven-Peak Trap	1	2	3	10K
B4	Equal Maxima	1	5	0	10K
B5	Decreasing Maxima	1	1	4	10K
B6	Uneven Maxima	1	5	0	10K
B7	Uneven Decreasing Maxima	1	1	4	10K
B8	Himmelblau	2	4	0	10K
B9	Six-Hump Camel Back	2	2	2	10K
B10	Shekel's Foxholes	2	1	24	10K
B11	Shubert	2	18	many	100K
B12	Vincent	1	6	0	20K
B13	Vincent	2	36	0	200K
B14	Vincent	3	216	0	400K

of three deceptive functions: Two-Peak Trap, Central Two-Peak Trap and Five-Uneven-Peak Trap. All have been expanded into the higher dimension number by using the simple formula:

$$y = \frac{\sum_{i=1}^D f(x_i)}{D} \quad (2)$$

The main difficulty introduced by those Deceptive Functions is a small niche area for the global optima and their location in the far 'corners' of the domain, where niches for the local optima are wide and located in the 'center' of the search space. Figure 2 illustrates selected function landscapes in 2D versions.

The Deceptive Functions test set has been evaluated using two budgets (see Table IV) for solving methods. The standard budget has been defined as more restrictive to create a challenge for the methods. However, the experiments have shown that so small number of births does not allow for convergence for any of the researched methods. Hence, in additional experiments extended budget has been used, where

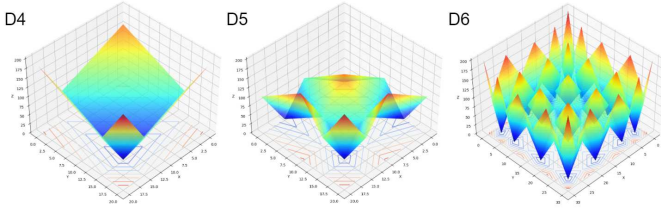


Fig. 2. **Deceptive functions visualization.** All three functions: Two-Peak Trap, Central Two-Peak Trap, Five-Uneven-Peak Trap in 2D variants.

TABLE IV
DECEPTIVE MULTIMODAL FUNCTION SET

#	Function Name	D	#GOPT	#LOPT	Budget	Budget+
D1	Two-Peak	1	1	1	10K	10K
D2	Central Two-Peak	1	1	1	10K	10K
D3	Five-Uneven-Peak	1	2	3	10K	10K
D4	Two-Peak	2	1	3	20K	40K
D5	Central Two-Peak	2	1	3	20K	40K
D6	Five-Uneven-Peak	2	4	21	20K	40K
D7	Two-Peak	3	1	7	40K	200K
D8	Central Two-Peak	3	1	7	40K	200K
D9	Five-Uneven-Peak	3	8	117	40K	200K

examined methods were given enough 'time' to converge and stabilize the results.

C. Results

To measure the efficiency of the examined method, two standard measures are used – PR and SR (defined in the previous section). In Table V the values of PR and SR of the GaMeDE and modified algorithms for the CEC2013 set are given. It is worth mentioning that scores for GaMeDE do not exactly match those achieved in the GECCO competition. The results are slightly diverse because of the non-deterministic character of GaMeDE but are within one standard deviation.

However, the average PR from all 20 problems maintains second place in the GECCO competition leader board. The GaMeDE2 managed to achieve comparable and even slightly better performance than GaMeDE. SDLCSDE algorithm results have not been found for this test set.

The values of PR and SR of all three methods for the Classical Functions set are given in Table VI. Results of GaMeDE and GaMeDE2 were calculated for the $\epsilon = 10^{-5}$ accuracy level, where those for SDLCSDE algorithm have been acquired from the original publication [4](see Table VIII) and with matching accuracy levels. Results show that for the functions B4, B5, B6, B7, B9, accuracy is higher (see Table VIII), but it is either the same or lower for the rest. To get more fair results, GaMeDE2 has been re-evaluated on functions B4–B9 with a matching accuracy which was presented in Table VII. Experiments on function 8 have been repeated due to lower performance than SDLCSDE while using higher accuracy. Thus, results show that GaMeDE2 achieves better results for B13 and B14 than SDLCSDE algorithm while maintaining a higher or the same accuracy level for all functions. Additionally, results show that GaMeDE struggles with a number of functions.

TABLE V
RESULTS FOR THE CEC2013 / GECCO2020 MULTIMODAL FUNCTION SET

#	GaMeDE		GaMeDE2		HVCMO	
	PR	SR	PR	SR	PR	SR
F1	1.000	1.000	1.000	1.000	1.000	1.000
F2	1.000	1.000	1.000	1.000	1.000	1.000
F3	0.967	0.967	1.000	1.000	1.000	1.000
F4	1.000	1.000	1.000	1.000	1.000	1.000
F5	1.000	1.000	1.000	1.000	1.000	1.000
F6	1.000	1.000	1.000	1.000	1.000	1.000
F7	1.000	1.000	1.000	1.000	1.000	1.000
F8	1.000	1.000	1.000	1.000	0.967	0.000
F9	1.000	1.000	1.000	1.000	0.937	0.000
F10	1.000	1.000	1.000	1.000	1.000	1.000
CF11	1.000	1.000	1.000	1.000	1.000	1.000
CF12	0.983	0.867	1.000	1.000	1.000	1.000
CF13	0.994	0.967	1.000	1.000	1.000	1.000
CF14	0.806	0.033	0.761	0.033	0.861	0.267
CF15	0.750	0.000	0.750	0.000	0.750	0.000
CF16	0.667	0.000	0.667	0.000	0.689	0.000
CF17	0.750	0.000	0.750	0.000	0.750	0.000
CF18	0.667	0.000	0.667	0.000	0.667	0.000
CF19	0.554	0.000	0.575	0.000	0.575	0.000
CF20	0.496	0.000	0.500	0.000	0.488	0.000
Avg stat	0.882	0.642	0.883	0.652	0.884	0.563
	=		=	++	=	

It is worth mentioning that GaMeDE2 results for the B8 instance have been achieved for the 0.00001 precision, while SDLCSDE has been evaluated for the 0.0005 precision. After re-evaluating (see Table VII) this instance on the same accuracy level, GaMeDE2 also achieved $PR = 1.0$ and $SR = 1.0$.

TABLE VI
RESULTS FOR THE CLASSICAL MULTIMODAL BENCHMARK FUNCTION SET.

#	GaMeDE [3]		GaMeDE2		HVCMO [6]		SDLCSDE [4]	
	PR	SR	PR	SR	PR	SR	PR	SR
B1	0.933	0.933	1.000	1.000	1.000	1.000	1.000	1.000
B2	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
B3	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
B4	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
B5	0.900	0.900	1.000	1.000	1.000	1.000	1.000	1.000
B6	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
B7	0.833	0.833	1.000	1.000	1.000	1.000	1.000	1.000
B8	0.433	0.000	0.992*	0.967*	1.000	1.000	1.000	1.000
B9	0.550	0.100	1.000	1.000	1.000	1.000	1.000	1.000
B10	0.067	0.067	1.000	1.000	0.967	0.967	1.000	1.000
B11	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
B12	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
B13	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.940
B14	1.000	1.000	1.000	1.000	0.938	0.000	0.889	0.000
Avg stat	0.837	0.774	1.000	1.000	0.993	0.926	0.992	0.924
			=	++	=			

Results of experiments in Table IX present GaMeDE, GaMeDE2 and HVCMO applications to the Deceptive Functions set. GaMeDE2 proved to achieve better results for every function except for D2 and D3 where both algorithms found all the solutions. Due to the novelty (and lack of SDLCSDE code) of this set there are no results for SDLCSDE.

In the methods evaluation process and experiments, there are some suggestions that for the Deceptive Multimodal Function Set it is worth extending the budget for several functions

TABLE VII
RESULTS FOR CLASSICAL MULTIMODAL BENCHMARK FUNCTION SET
– REEVALUATION

#	Accuracy	PR	SR
B4	0.000001	1.000	1.000
B5	0.000001	1.000	1.000
B6	0.000001	1.000	1.000
B7	0.000001	1.000	1.000
B8	0.0005	1.000	1.000
B9	0.000001	1.000	1.000

TABLE VIII
ACCURACY LEVELS FOR SDLCSDE [4]

#	Accuracy
B1	0.05
B2	0.05
B3	0.05
B4	0.000001
B5	0.000001
B6	0.000001
B7	0.000001
B8	0.0005
B9	0.000001
B10	0.00001
B11	0.05
B12	0.0001
B13	0.001
B14	0.001

(especially for D7-D9) to explore the three-dimensional landscape more extensively. The standard and extended budgets are presented in Table IV. In Table X the summary of results for examined methods is presented.

Table X includes results with extended budgets for all examined methods. It can be concluded that each method uses the extended budget effectively and achieves better results. However, GaMeDE2 is the most effective: $PR = 1.0$ and $SR = 1.0$, as it solves each function and outperforms other methods.

To summarise, the results for three test sets: CEC2013, Classic and Deceptive are presented in Table XI. Presented data show the average values for PR and SR for four examined methods: GaMeDE, GaMeDE2, HVCMO and SDLCSDE. Results show that GaMeDE2 outperforms referenced methods.

The results presented in this section contain four methods and three test sets. It gives a rather bird’s eye view of examined methods. More detailed results, analysis and discussion, are included in the next section.

D. Discussion

The GaMeDE2 maintained the average $PR = 0.883$ ($W = 6 > W_{\rho < 0.05}$) on the CEC2013 (see Table V) test set but improved the SR for F3, F12, F13. CF14 is the only instance where PR value decreased, but the improvement on the remaining ones has balanced it. It was not expected to observe any significant change in this benchmark set used for the original GaMeDE method development. On the other hand, it is a crucial result suggesting that proposed modifications do not harm any original components. Difference between GaMeDE2 and HVCMO $PR = 0.884$ is not a significant

TABLE IX
RESULTS FOR THE DECEPTIVE MULTIMODAL FUNCTION SET – STD.
BUDGETS

#	GaMeDE		GaMeDE2		HVCMO	
	PR	SR	PR	SR	PR	SR
D1	0.900	0.900	1.000	1.000	1.000	1.000
D2	1.000	1.000	1.000	1.000	1.000	1.000
D3	1.000	1.000	1.000	1.000	1.000	1.000
D4	0.067	0.067	1.000	1.000	1.000	1.000
D5	0.167	0.167	1.000	1.000	1.000	1.000
D6	0.642	0.133	0.867	0.600	0.992	0.967
D7	0.333	0.333	1.000	1.000	1.000	1.000
D8	0.133	0.133	1.000	1.000	1.000	1.000
D9	0.238	0.000	0.267	0.000	0.479	0.033
Avg stat	0.498	0.415	0.904	0.844	0.941	0.889
					++	++

TABLE X
RESULTS FOR THE DECEPTIVE MULTIMODAL FUNCTION SET – EXT.
BUDGETS

#	GaMeDE		GaMeDE2		HVCMO	
	PR	SR	PR	SR	PR	SR
D1	0.900	0.900	1.000	1.000	1.000	1.000
D2	1.000	1.000	1.000	1.000	1.000	1.000
D3	1.000	1.000	1.000	1.000	1.000	1.000
D4	0.100	0.100	1.000	1.000	1.000	1.000
D5	0.367	0.367	1.000	1.000	1.000	1.000
D6	1.000	1.000	1.000	1.000	1.000	1.000
D7	0.667	0.667	1.000	1.000	1.000	1.000
D8	0.567	0.567	1.000	1.000	1.000	1.000
D9	0.554	0.233	1.000	1.000	0.996	0.967
Avg stat	0.684	0.648	1.000	1.000	1.000	0.996
			=	=	=	=

difference either ($W = 7 > W_{\rho < 0.05}$). In comparison to the HVCMO method, the key advantage of the GaMeDE2 method is a SR difference, especially for functions F8 and F9, where it found all optima in every run, while HVCMO failed to do so even once. These two functions’ main feature is the uneven distribution of high optima number (same as B13 - see Figure 5). Later in this chapter, it is explained which element is responsible for this improvement.

The second test set, containing Classical Multimodal Benchmark Functions (see Table VI), introduces novel instances for the GaMeDE. While some of the functions are repeated, they have narrowed the evaluation budget. Plots in Figure 3 visualize that selection in the first iteration doubles the initial evaluation cost, which is the 80% for both B8 and B10. With only 20% budget left, it is unlikely for the algorithm to locate all global optima. Skipping the first selection proposed as the first modification reduced the initial cost to 40% of the total budget.

Moreover, to fully take advantage of the extended evaluation budget, the clustering procedure in GaMeDE2 has been replaced with the initial iteration. In GaMeDE, the current population is clustered around the set of archived points. Those archive points are selected to reward unexplored areas. It is a valuable mechanism. However, there is no application in the first iteration, where the whole area is unexplored. The

TABLE XI
SUMMARY RESULTS FOR CEC2013, CLASSIC AND DECEPTIVE SETS

Set	GaMeDE		GaMeDE2		HVCMO		SDLCSDE	
	PR	SR	PR	SR	PR	SR	PR	SR
CEC2013	0.882	0.642	0.883	0.652	0.884	0.563	-	-
Classic	0.837	0.774	1.000	1.000	0.993	0.926	0.992	0.924
Deceptive (std)	0.498	0.415	0.904	0.844	0.941	0.889	-	-
Deceptive (ext)	0.684	0.648	1.000	1.000	1.000	0.996	-	-

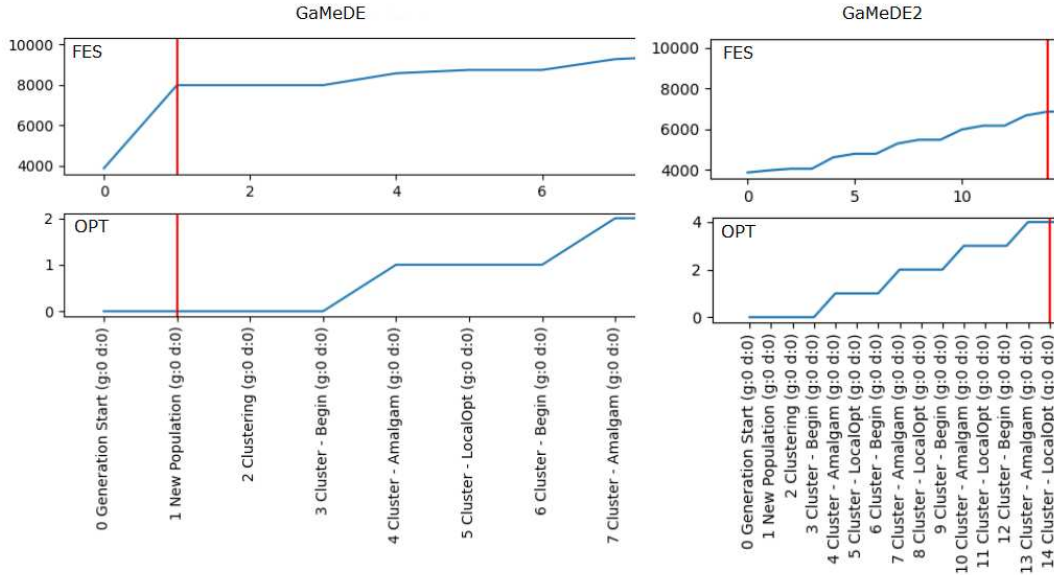


Fig. 3. **Modification #1 improvement.** Budget used (FES) and Optima found (OPT) for B8 problem instance before and after skipping selection in initial iteration. Similar effect can be observed for the instance B10.

alternative version introduces additional clustering of archive points just before the current population is clustered around them. It allows for filtering out archive points if there is a probability that they lay in the same niche. The modification results are presented on the Figure 4.

While having the explicit *WIDE* and *FOCUS* phases is not crucial for GaMeDE2, it is still important to maintain both types of selection: *GAP* and *Random*. The important difference is that the former is archive-based, while the latter is population-based. Based on the experiments, both selections proved to be crucial. Using solely *Random* Selection gives comparable results for most of the instances, except for those with a high number of optima (F8, F9, B13, B14), where method struggled with finding the narrow optima in far corners of the search space. On the contrary, while using *GAP* selection only, all the optima in mentioned instances have been found. Figure 5 presents the difference in population distribution after using *Random* or *GAP* selection. A drawback of this approach has been however observed in decreased *PR* for high-dimensional instances (CF19, see Table V). It confirms that using both alternately gives the best results. Further research could support further simplification of the method by limiting to *GAP* selection only. Statistical tests for the Classical Functions Set (see Table VI) confirm all these

changes introduce a significant improvement of $PR = 1.0$ ($W = 0 \leq W_{\rho < 0.05}$) and $SR = 1.0$ ($W = 0 \leq W_{\rho < 0.05}$) over the original GaMeDE method. It allowed achieving similar effectiveness to the HVCMO method.

The proposed Deceptive Multimodal Function Set introduces Trap functions in two- and three- dimensions. The essential difficulty is the moderate budget scaling which creates a challenge in three-dimension variants. Another difficulty is the very steep global optima in the far corners of the search space opposite the large area of deceptive local optima. The GaMeDE struggles with those functions because a significant fraction of the archive points lies on the local optima, and there is a bigger chance of selecting them for optimization. GaMeDE2 has been statistically verified to improve the $PR = 1.0$ ($W = 0 \leq W_{\rho < 0.05}$) and $SR = 1.0$ ($W = 0 \leq W_{\rho < 0.05}$) in relation to the original GaMeDE method for both budget sets (see Table IX and Table X).

V. CONCLUSIONS AND FUTURE WORK

Developing a method for a specific benchmark suite allows to focus on the improvements and quickly verify their effectiveness. However, it may lead to overfitting of the proposed solution. The original GaMeDE could be such a case. While very competitive on the CEC2013 benchmarks,

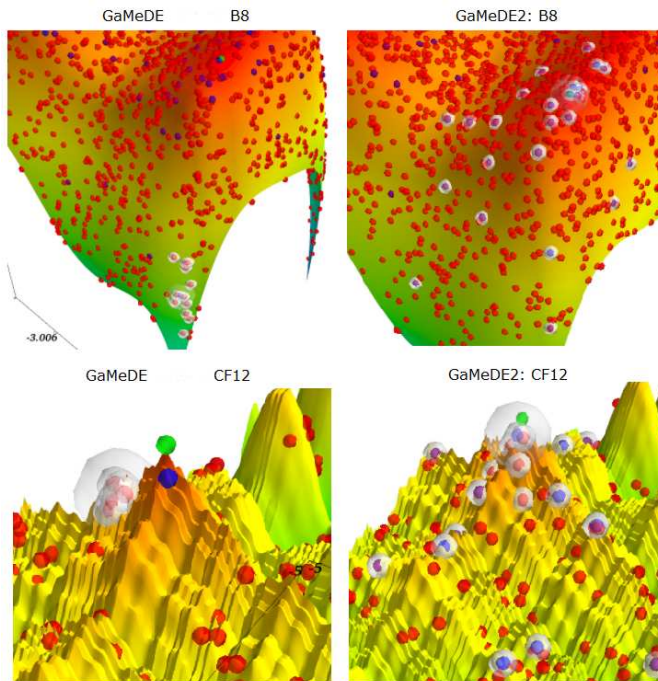


Fig. 4. **Modification #3 improvement.** After additional archive points clustering, the actual clusters (marked by grey spheres) and located around the global optima (green spheres) instead of on their side.

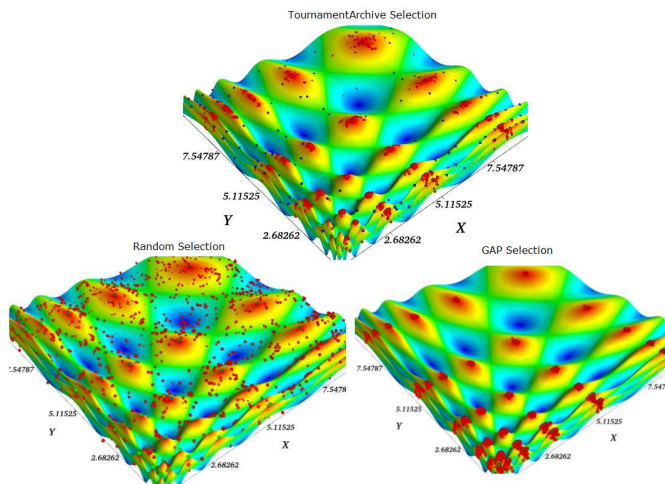


Fig. 5. **Random and GAP selection effects (B13 func.)** population distribution (red spheres) after Random or GAP selection

it does not maintain the effectiveness on novel instances. Two additional test sets have been proposed to verify the method’s generalization ability. First, two deceptive functions (Two-Peak Trap, Central Two-Peak Trap), Decreasing Maxima, Uneven Maxima, and Shekel’s Foxholes have been added. Additionally, the computational budget for the number of already present functions has been limited. The second proposed test set further explores the range of the deceptive function – it expands three trap functions into 2 and 3 dimensions.

The Hill-Valley-Clustering-based VMO (HVcMO), a novel solution based on the HillValIEA, has been selected to provide a fair comparison.

In tuning procedure GaMeDE2, the Taguchi parameter design procedure has been used to evaluate the GaMeDE parameters and fine-tune them across all three sets. While it provided a promising outcome for a single instance, it could not point a one configuration valid for all instances of the test suites. Such an outcome is that specific instances introduce different challenges, sometimes overlapping each other. For example, Vincent function has strongly irregularly distributed yet smooth optima. Composition Functions such as CF17/CF20 consist of two steep optima. Shekel’s Foxholes is a nearly binary landscape with a minimal variety among the optima. Furthermore, there is a wide span of provided budget, even for the same functions: 50K evaluations for Himmelblau in F4 and only 10K in B8. Decreasing the budget might expose the weak spots of methods that otherwise work successfully. Fine-tuning parameters per instance ensures the best results yet requires far more studies of the problem and time. However, the number of optima, their distribution, and local landscape disturbance have not been known *a priori*. For all those reasons, having a single configuration could be a superior approach, and optimization methods that do not require tuning many parameters are far more practical.

Based on the results of the experiments, a set of changes has been proposed to improve the original method’s generality, leading to better results for the two novel test sets. Moreover, it maintains the CEC2013 benchmark functions’ competitive level while using just a single parameter configuration. In comparison to another state-of-the-art method – HVcMO, GaMeDE2 manages to completely solve instances with a high number of unevenly distributed optima (F8, F9, B14), while maintaining the average *PR* at the same level. The only case where introduces method has lower effectiveness is a novel Deceptive Functions Set if a very restrictive budget is provided.

The proposed GaMeDE2 method mainly addresses the initialization and clustering process. Skipping the initial mutation frees a significant amount of budget, which prevents premature algorithm stopping for the low-budget instances. The alternative (double) clustering in the first iteration allows for faster exploring all the promising niches. While it has low chances of finding narrow optima, it marks the ‘easy’ ones. The idea is similar to the mechanism of two natural metabolism phases (*Anaerobic* and *Aerobic*) in the human body. The first is used in short, burst activity (a big number of ‘easy’ optima). The second can be for long-duration activities and far goals (narrow, ‘hard’ steep optima). Additionally, GaMeDE2 is further simplified by removing the selection dependency from the phase.

The proposed Deceptive Functions set illustrates that Trap functions are not a bigger challenge for the multimodal solving methods. Multimodal optimization, by definition, does not seek a sole solution, which makes it more resistant to deceptiveness. Further research on the more irregular high-

dimensional composition of different Trap Functions could be valuable for further research direction.

Though GaMeDE2 uses a single configuration, its further version could also benefit from adaptive parameters steering, such as population size, archive size, mutation or crossover probability. Moreover, simplifying the original method, e.g. the parameter used in enabling *HillClimbing* step was not fully eliminated – it requires further work to make it fully adaptive. Indeed, improvements in clustering could be a promising research area. At the current state, the first iteration allows searching all the promising niches, leading to extensive use of the evaluation budget. An efficient mechanism in balancing and prioritizing the clusters to explore could introduce a significant value.

REFERENCES

- [1] S. Das, S. Maity, B.-Y. Qu, P. N. Suganthan, Real-parameter evolutionary multimodal optimization—a survey of the state-of-the-art, *Swarm and Evolutionary Computation* 1 (2) (2011) 71–88.
- [2] X. Li, A. Engelbrecht, M. G. Epitropakis, Benchmark functions for cec'2013 special session and competition on niching methods for multimodal function optimization, RMIT University, Evolutionary Computation and Machine Learning Group, Australia, Tech. Rep.
- [3] M. Laszczyk, P. B. Myszkowski, A gap-based memetic differential evolution (gamede) applied to multi-modal optimisation—using multi-objective optimization concepts, in: *Intelligent Information and Database Systems: 13th Asian Conference, ACIIDS 2021, Phuket, Thailand, April 7–10, 2021, Proceedings 13*, Springer International Publishing, 2021, pp. 211–223.
- [4] Q. Liu, S. Du, B. J. van Wyk, Y. Sun, Double-layer-clustering differential evolution multimodal optimization by speciation and self-adaptive strategies, *Information Sciences* 545 (2021) 465–486.
- [5] X. Li, Niching without niching parameters: particle swarm optimization using a ring topology, *IEEE Transactions on Evolutionary Computation* 14 (1) (2009) 150–169.
- [6] R. Navarro, C. H. Kim, Niching multimodal landscapes faster yet effectively: Vmo and hillvallea benefit together, *Mathematics* 8 (5) (2020) 665.
- [7] S. Maree, T. Alderliesten, D. Thierens, P. A. Bosman, Real-valued evolutionary multi-modal optimization driven by hill-valley clustering, in: *Proceedings of the genetic and evolutionary computation conference*, 2018, pp. 857–864.
- [8] S. W. Mahfoud, Niching methods for genetic algorithms, Ph.D. thesis, University of Illinois at Urbana-Champaign (1995).
- [9] J. Barrera, C. A. C. Coello, *A Review of Particle Swarm Optimization Methods Used for Multimodal Optimization*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2009, pp. 9–37.
- [10] B.-Y. Qu, P. N. Suganthan, Novel multimodal problems and differential evolution with ensemble of restricted tournament selection, in: *IEEE Congress on Evolutionary Computation, IEEE*, 2010, pp. 1–7.
- [11] D. E. Goldberg, K. Deb, J. Horn, Massive multimodality, deception, and genetic algorithms., in: *PPSN*, Vol. 2, 1992.
- [12] B.-Y. Qu, J. J. Liang, Z. Wang, Q. Chen, P. N. Suganthan, Novel benchmark functions for continuous multimodal optimization with comparative results, *Swarm and Evolutionary Computation* 26 (2016) 23–34.
- [13] B.-Y. Qu, J. J. Liang, P. N. Suganthan, Niching particle swarm optimization with local search for multi-modal optimization, *Information Sciences* 197 (2012) 131–143.
- [14] S. Yazdani, H. Nezamabadi-pour, S. Kamyab, A gravitational search algorithm for multimodal optimization, *Swarm and Evolutionary Computation* 14 (2014) 1–14.
- [15] K. Price, R. M. Storn, J. A. Lampinen, *Differential evolution: a practical approach to global optimization*, Springer Science & Business Media, 2006.
- [16] Z.-J. Wang, Z.-H. Zhan, Y. Lin, W.-J. Yu, H.-Q. Yuan, T.-L. Gu, S. Kwong, J. Zhang, Dual-strategy differential evolution with affinity propagation clustering for multimodal optimization problems, *IEEE Transactions on Evolutionary Computation* 22 (6) (2017) 894–908.
- [17] H. Zhao, Z.-H. Zhan, Y. Lin, X. Chen, X.-N. Luo, J. Zhang, S. Kwong, J. Zhang, Local binary pattern-based adaptive differential evolution for multimodal optimization problems, *IEEE transactions on cybernetics* 50 (7) (2019) 3343–3357.
- [18] Z.-G. Chen, Z.-H. Zhan, H. Wang, J. Zhang, Distributed individuals for multiple peaks: A novel differential evolution for multimodal optimization problems, *IEEE Transactions on Evolutionary Computation* 24 (4) (2019) 708–719.
- [19] P. Haghbayan, H. Nezamabadi-Pour, S. Kamyab, A niche gsa method with nearest neighbor scheme for multimodal optimization, *Swarm and evolutionary computation* 35 (2017) 78–92.
- [20] Y. Li, Y. Chen, J. Zhong, Z. Huang, Niching particle swarm optimization with equilibrium factor for multi-modal optimization, *Information Sciences* 494 (2019) 233–246.
- [21] Z.-J. Wang, Z.-H. Zhan, Y. Lin, W.-J. Yu, H. Wang, S. Kwong, J. Zhang, Automatic niching differential evolution with contour prediction approach for multimodal optimization problems, *IEEE Transactions on Evolutionary Computation* 24 (1) (2019) 114–128.
- [22] S. Maree, T. Alderliesten, P. A. Bosman, Benchmarking hillvallea for the gecco 2019 competition on multimodal optimization, *arXiv preprint arXiv:1907.10988*.
- [23] R. K. Ursem, Multinational evolutionary algorithms, in: *Proceedings of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406)*, Vol. 3, IEEE, 1999, pp. 1633–1640.
- [24] A. Puris, R. Bello, D. Molina, F. Herrera, Variable mesh optimization for continuous optimization problems, *Soft Computing* 16 (3) (2012) 511–525.
- [25] P. A. Bosman, J. Grahl, D. Thierens, Benchmarking parameter-free amalgam on functions with and without noise, *Evolutionary computation* 21 (3) (2013) 445–469.
- [26] V. N. Nair, B. Abraham, J. MacKay, G. Box, R. N. Kacker, T. J. Lorenzen, J. M. Lucas, R. H. Myers, G. G. Vining, J. A. Nelder, et al., Taguchi's parameter design: a panel discussion, *Technometrics* 34 (2) (1992) 127–161.

A chance-constraint approach for optimizing Social Engagement-based services

Michel Bierlaire*, Edoardo Fadda[†], Lohic Fotio Tiotso[‡], and Daniele Manerba[§]

*Transport and Mobility Laboratory, Ecole Polytechnique Fédérale de Lausanne - CH-1015 EPFL, Lausanne (Switzerland)

[†]Dept. of Mathematical Sciences, Politecnico di Torino - corso Duca degli Abruzzi 24, Torino (Italy)

[‡]Dept. of Control and Computer Engineering, Politecnico di Torino - corso Duca degli Abruzzi 24, Torino (Italy)

[§]Dept. of Information Engineering, Università degli Studi di Brescia, via Branze 38, Brescia (Italy)

Emails: michel.bierlaire@epfl.ch, edoardo.fadda@polito.it, lohic.fotiotiotso@polito.it, daniele.manerba@unibs.it

Abstract—Social Engagement is a novel business model transforming final users of a service from passive into active components. In this framework, people are contacted by a company and they are asked to perform tasks in exchange for a reward. This arises the complicated optimization problem of allocating the different types of workforce so as to minimize costs. We address this problem by explicitly modeling the behavior of contacted candidates through consolidated concepts from utility theory and proposing a chance-constrained optimization model aiming at optimally deciding which user to contact, the amount of the reward proposed, and how many employees to use in order to minimize the total expected costs of the operations. A solution approach is proposed and its computational efficiency is investigated through experiments.

I. INTRODUCTION AND RELATED WORKS

SOcial Engagement (SE) is a new business paradigm involving the customers of a company in its operations. More precisely, people agree to perform specific services in exchange for a reward. This model has been enabled by the increase of the number of users connected on the web and technologies able to get people information [1]. This gives to the companies the possibility to easily communicate with *candidates* and then to propose *tasks* in exchange for a reward.

A concrete application of the SE paradigm is the so called *crowd-shipping* logistics, in which the companies ask the people to collect the packages to a certain location and deliver it to the final user [2][3]. By doing this, companies do not only decrease the costs, but also the environmental impact since people accepting the delivery usually would take advantage of travels that they have to do anyhow for other activities. Another interesting application of SE occurs in an evolution of the Internet of Things (IoT) concept called opportunistic IoT (oIoT) [4]. Since the IoT development is considerably slowed down by the difficulty and costs involved in building telecommunication networks capable of continuously transmitting large amounts of data collected by sensors, through oIoT the citizens share (in exchange for a reward) the internet of their devices (mobile phones, modems) so that the nearby sensors can exploit it to communicate the gathered data. In this work, we do not want to concentrate on a specific application rather on a very general SE-based setting in order to embrace all the basic characteristics of such a business model.

An effective planning of operations under the SE paradigm yields an interesting optimization problem. The decision-maker must decide how much he is willing to pay to a candidate for each task, when and where to rely on employees and on candidates, which tasks to assign to the employees and for which tasks the candidates must be contacted, in order to minimize the total operational costs. It is important to note that the reward paid to a candidate is generally lower on average than the cost that the company bears for an employee. However, while an employee is obliged to accept and carry out the tasks assigned to him, there is no certainty that a candidate will accept a proposed task.

Little attention has been devoted to the development of optimization models aimed at effectively scheduling companies operations that exploit SE. Just few works [5][6][7] have tried to tackle the problem and, therefore, there is a large room for improvement of existing approaches as well as for the design of more innovative and complete ones (as claimed regarding crowd-shipping in [3]). In particular, to the best of our knowledge, there is no published optimization model that explicitly accounts for individual candidate behaviour when planning SE-based operations. As already mentioned, one characteristic that makes challenging the optimization problems deriving from the implementation of the SE paradigm is the fact that candidates are not constrained a priori to respect a contract. This means that, once contacted, the candidate may not accept the task and, if we assume a pure rational profit-maximization behavior of the candidate, the reject can happen because the proposed reward is lower than the candidate expectation. It is therefore important to integrate tools in the decision-making process that allow monitoring the individual behavior of potential candidates.

In this work, to account for individual behaviour, we rely on the candidate's *willingness to accept* (wta) a task, i.e., the minimum reward expected by a candidate to accept a task. The wta is a well consolidated concept in utility theory and has been used since long to explain human subject preferences in economics [8]. From the decision-maker point of view, the candidate's wta is not deterministically known, since it depends on some factors that are intrinsic of the candidates. Therefore, we consider the candidate wta as a random variable.

Thus, the probability of acceptance for a candidate will be equal to the probability that the offered reward is greater or equal to the wta of the candidate. The adopted perspective is similar to [6]. However, instead of relying on a single random variable describing the number of candidates, we model each single candidate behavior through a Bernoulli random variable. The parameter of such a Bernoulli random variable, i.e. the probability that the candidate accepts the task, is not fixed but depends by the proposed reward.

This paper's contribution is twofold. First, we propose a novel mathematical model for SE-based services optimization. The formulation, which includes chance constraints [9], results to be the first one that explicitly accounts for each individual candidates behaviour. Second, since the complexity of the proposed model and the explicit consideration of stochastic parameters do not allow to obtain a simple solution, we derive a mixed-integer quadratic programming model that approximates the original model. This is done by making some reasonable hypothesis on the probability distribution of the wta of each candidate, and by exploiting the Markov inequality. Several computational experiments validate the suitability of our proposed model and solution approach.

The rest of the paper is organized as follows. The optimization problem is defined and modeled in Section II. Our solution approach is described in Section III. Section IV presents the experimental results, while Section V concludes the paper.

II. THE SOCIAL ENGAGEMENT OPTIMIZATION PROBLEM

The social engagement optimization problem that we want to study considers a decision-maker (in general a company) whose goal is to use people, in the following called *candidate*, in addition to employees in order to perform a set of tasks. In particular, we consider a urban environment divided in several geographical areas such as mobile phone cells, neighborhoods of different markets or just geographical areas. Each of these areas is characterized by a number of tasks to perform and each task is characterized by different workloads, thus a single task may require more candidates to be done. For example, in the crowd shipping setting these tasks are the delivery required by customers out of the store, while in the oIoT application these tasks consist in sharing the internet connection with smart sensors in the city.

Each task can either be performed by using employees or candidates. Employee are more expensive, are available in a small number but they execute the tasks assigned. Instead, candidates are less expensive, their quantity is virtually unlimited (since the number of people considered for SE is far greater than the number of tasks) but they can refuse to perform a task with a given probability. We assume that the acceptance probability increases as the offered reward increase. Please note that, in practice, an employee has greater productivity than a candidate. The goal of the decision-maker is to minimize the total operative costs while enforcing that with high probability all the tasks must be performed.

Let us consider a set \mathcal{I} of tasks and a set \mathcal{M} of candidates. For each task i , let W_i be the workload required, α_i be the

required probability for its accomplishment, Δ_i^m be a random variable representing the wta of candidate m , and c_i be the cost of using an employee. Moreover, let B be the number of available employees and $r > 1$ be the ratio between the productivity of an employee and that of a candidate, i.e., the workload that a single employee can afford as compared to a candidate in the same time frame.

We define the decision variables $Q_i^m \in \mathbb{R}^+$ as the reward offered to candidate m to accept task i and $z_i \in \mathbb{N}$ as the number of employee assigned to tasks i . Moreover, we consider the probability for candidate m to accept task i called $x_i^m \in [0, 1]$ and the random variables Y_i^m distributed according to a Bernoulli distribution of probability x_i^m which assume value 1 if candidate m accepts to perform task i . Then, the Social Engagement Optimization Problem (SEOP) can be formulated as follows:

$$\min \sum_{i \in \mathcal{I}} \sum_{m \in \mathcal{M}} Q_i^m x_i^m + \sum_{i \in \mathcal{I}} c_i z_i \quad (1)$$

$$\text{s.t. } x_i^m = \mathbb{P}[Q_i^m \geq \Delta_i^m], i \in \mathcal{I}, m \in \mathcal{M} \quad (2)$$

$$\mathbb{P}[Y_i^m = a] = (x_i^m)^a (1 - x_i^m)^{(1-a)}, i \in \mathcal{I}, m \in \mathcal{M} \quad (3)$$

$$a \in \{0, 1\} \quad (4)$$

$$\mathbb{P} \left[\sum_{m \in \mathcal{M}} Y_i^m + r z_i \geq W_i \right] \geq \alpha_i, i \in \mathcal{I} \quad (5)$$

$$\sum_{i \in \mathcal{I}} z_i \leq B \quad (6)$$

$$z_i \in \mathbb{N}, i \in \mathcal{I}, \quad (7)$$

$$Q_i^m \in \mathbb{R}^+, x_i^m, Y_i^m \in [0, 1], i \in \mathcal{I}, m \in \mathcal{M}. \quad (8)$$

The total cost in (1) is expressed as the summation between the total expected cost offered as rewards (the reward Q_i^m is paid with probability x_i^m), and the sum of the costs paid for employees. Constraints (2) define the variables x_i^m as the acceptance probability, while constraints (3) and (4) ensure Y_i^m to follow a Bernoulli distribution. Constraints (5) are chance constraints enforcing a minimum probability of doing a given task either by using employees or candidates. It is worth noting that ensuring that each task is performed with a given probability is less strict than requiring that all the tasks will be performed with a given probability. Nevertheless enforcing this second condition would lead to too conservative solutions. Finally, constraint (6) accounts for the limited number of employees.

III. SOLUTION APPROACH

The optimization problem in (1)-(6) is difficult to solve due to the definition of x_i^m in constraints (2), of Y_i^m in constraints (3) and (4), and the chance constraints in (5). Hence, we approximate these constraints in order to get a model which can be readily solved with off-the-shelf solvers.

Constraints (2) involve the cdf of the random variable Δ_i^m . We approximate it by means of a piece-wise linear function with J breakpoints. In particular, instead of constraints (2) we add a set of constraints of the form

$$x_i^m \leq k_j Q_i^m + q_j, \quad j = 1, \dots, J, i \in \mathcal{I}, m \in \mathcal{M}, \quad (9)$$

where k_j and q_j are obtained by imposing proper conditions (e.g. the passage in J points of the cdf). This choice is equivalent to enforce $x_i^m \leq \min[1, m_1 Q_i^m + q_1, \dots, m_J Q_i^m + q_J]$, where the first term of the minimum comes from the definition of x_i^m . Since the approximation proposed in (10) just lead to concave functions (being the pointwise minimum of affine functions) and since the a general cdf may be convex in some portion of the domain, the proposed approximation is not guarantee to converge to the cdf for all the distributions. In the following, for the sake of simplicity, we consider just $J = 1$ and we impose the passage for the point $(0, 0)$ meaning that with 0 reward the probability that the candidate will perform the task is 0, and for the point $(\bar{Q}_i^m, 1)$ where \bar{Q}_i^m is a reward for which the candidate m is willing to perform the task i with a probability that we may approximate to be 1. By making this choice, the obtained final approximation of Constraints (2) is:

$$x_i^m \leq Q_i^m / \bar{Q}_i^m, \quad i \in \mathcal{I}, m \in \mathcal{M}. \quad (10)$$

Now let us consider the constraints in (5), note that these constraints can be written as:

$$\mathbb{P} \left[\sum_{m \in \mathcal{M}} Y_i^m \geq W_i - rz_i \right] \geq \alpha_i, \quad i \in \mathcal{I}. \quad (11)$$

By using the Markov inequality, for each $i \in \mathcal{I}$, it holds:

$$\frac{\mathbb{E}[\sum_{m \in \mathcal{M}} Y_i^m]}{W_i - rz_i} \geq \mathbb{P} \left[\sum_{m \in \mathcal{M}} Y_i^m \geq W_i - rz_i \right] \geq \alpha_i. \quad (12)$$

Hence, since $\mathbb{E}[\sum_{m \in \mathcal{M}} Y_i^m] = \sum_{m \in \mathcal{M}} \mathbb{E}[Y_i^m] = \sum_{m \in \mathcal{M}} x_i^m$, Eq. (12) leads to the following constraint:

$$\sum_{m \in \mathcal{M}} x_i^m \geq \alpha_i (W_i - rz_i), \quad i \in \mathcal{I}. \quad (13)$$

Eq. (13) is enforcing that the expected workload form the candidates must be greater than the α_i percent of the people needed. Moreover, by considering the bound provided by Eq. (13), we are reducing the feasible set, thus the condition in (11) will be satisfied for greater value of α_i .

Then, the resulting approximation of the SEOP ($SEOP_{ap}$) is the following mixed integer quadratic model:

$$\min \sum_{i \in \mathcal{I}} \sum_{m \in \mathcal{M}} Q_i^m x_i^m + \sum_{i \in \mathcal{I}} c_i z_i \quad (14)$$

$$\text{s.t. } x_i^m \leq \frac{Q_i^m}{\bar{Q}_i^m}, \quad i \in \mathcal{I}, m \in \mathcal{M} \quad (15)$$

$$\sum_{m \in \mathcal{M}} x_i^m \geq \alpha_i (W_i - rz_i), \quad i \in \mathcal{I} \quad (16)$$

$$\sum_{i \in \mathcal{I}} z_i \leq B \quad (17)$$

$$z_i \in \mathbb{N}, \quad i \in \mathcal{I}, \quad (18)$$

$$Q_i^m \in \mathbb{R}^+, x_i^m \in [0, 1], \quad i \in \mathcal{I}, m \in \mathcal{M}. \quad (19)$$

IV. EXPERIMENTAL RESULTS AND ANALYSIS

We now present CPU experiments validating the proposed solution approach. All the experiments were performed on a *Intel(R) Core(TM) i7-5500U CPU@2.40GHz* computer with 16GB of RAM and running *Ubuntu v20.04*. The exact solver used was Gurobi v9.1.1 via its Python3 APIs. The instances, according to realistic crowd-shipping scenarios, were generated considering $|\mathcal{I}| = \{5, 10, 20, 50, 100\}$, $|\mathcal{M}| = 4|\mathcal{I}|$, $w_i = 7 * (0.8 + 0.4 * \mathcal{U}(0, 1))$, $B = 2$, and $\alpha_i = \bar{\alpha}, \forall i \in \mathcal{I}$ with $\bar{\alpha}$ drawn from $\mathcal{U}(0.6, 0.9)$. Finally, the random variables Δ_i^m were drawn from a Gumbel distribution, while \bar{Q}_i^m was set to be equal to the 99 percentile of Δ_i^m .

A. CPU results

We first study the CPU solving time with respect to the dimension of the $SEOP_{ap}$. In particular, versus the growth of $|\mathcal{I}|$ and $|\mathcal{M}|$, we evaluate the CPU time (sec), the time-to-best (sec) (the number of seconds from the start of the execution of Gurobi to the time in which it finds the best solution of the run), and the MIP gap (%) (computed as the percentage difference between the lower and upper objective bound. In particular, we consider the least gap value that Gurobi has to reach before stopping its execution). The average and standard deviation on 10 instances are shown in Table I. In all the runs we set the solver time limit to 1 hour.

TABLE I
AVERAGE $[\mu]$ AND STANDARD DEVIATIONS $[\sigma]$ OF THE CPU TIME, TIME TO BEST, AND MIP GAP FOR DIFFERENT VALUES OF \mathcal{I} AND \mathcal{M} .

Instance		CPU time(sec)		time-to-best (sec)		MIP gap (%)	
$ \mathcal{I} $	$ \mathcal{M} $	μ	σ	μ	σ	μ	σ
5	20	0.09	0.01	0.09	0.01	0	0
10	40	1569.56	175.83	614.31	235.43	0	0
20	80	2780.84	573.26	2634.94	897.98	0	0
50	200	3600.00	0.00	1054.31	562.25	56	47
100	400	3600.00	0.00	1679.57	720.77	100	0

Instances with $|\mathcal{I}| = 5$, and $|\mathcal{M}| = 20$ are solved almost instantaneously with 0 gap. The time-to-best is equal to the CPU time since the difference are below the hundredths of a second. For instances with $|\mathcal{I}| = 10, |\mathcal{M}| = 40$, and $|\mathcal{I}| = 20, |\mathcal{M}| = 80$, the CPU time increases, but the solver is still able to find the optimal solution inside the time limit. For the instances with $|\mathcal{I}| = 10, |\mathcal{M}| = 40$ the time to best is near one half of the total CPU time but solutions with gap below the 5% are found by the solver already in the first minutes of the run. Instead, for the instance with $|\mathcal{I}| = 20, |\mathcal{M}| = 80$, the time-to-best is close to the whole computation time and no solution with gap below the 5% is found in the first minute of the run. For instances of greater dimensions, the solver is not able to find the optimal solution in the given time limit, for this reason the CPU time is equal to 3600 seconds with a standard deviation of 0. Nevertheless, for instances with $|\mathcal{I}| = 50$, and $|\mathcal{M}| = 200$, several times, the final gaps are smaller than 10%, while for instances with $|\mathcal{I}| = 100$, and $|\mathcal{M}| = 400$, the solver is not able to find a good bound in the allocated computational time, hence, a 100% MIP gap with 0 standard deviation is reported.

B. Approximation analysis

We now analyze the goodness of the $SEOP_{ap}$ approximation. Since Δ_i^m is distributed as a Gumbel distribution with concave cdf, the approximation proposed converges to the exact function and several techniques for developing good piece-wise approximation are available [10]. Thus, we are interested in quantifying how much conservative is the Markov inequality with respect to Eq. (5). Hence, we compute the optimal solution of $SEOP$ and we use it to calculate $\hat{\alpha} := \mathbb{P}[\sum_{m \in \mathcal{M}} Y_i^m + rz_i \geq W_i]$. This can be done easily by noting that the Y_i^m are independent with respect to the index m since the knowledge about candidate m performing a task does not provide any information related to the execution of the same task by other candidates. Thus, $\sum_m Y_i^m$ is a sum of independent random variable distributed according to Bernoulli distribution of parameter x_i^m . Central Limit Theorems for non identically distributed random variables are available and, in particular, by applying the Lyapunov Central Limit Theorem it is possible to prove [11] that for large values of $|\mathcal{M}|$ (in practice $|\mathcal{M}| \geq 30$), it holds that:

$$\sum_{m \in \mathcal{M}} Y_i^m \sim \mathcal{N} \left(\sum_{m \in \mathcal{M}} x_i^m, \sum_{m \in \mathcal{M}} x_i^m(1 - x_i^m) \right), \quad i \in \mathcal{I}. \quad (20)$$

By using (20), we can compute α by solving:

$$\alpha_i = 1 - \Phi \left(\frac{W_i - rz_i - \sum_{m \in \mathcal{M}} x_i^m}{\sqrt{\sum_{m \in \mathcal{M}} x_i^m(1 - x_i^m)}} \right), \quad i \in \mathcal{I}, \quad (21)$$

where Φ is the cdf of a standard normal distribution. We report the value of the $\alpha \in [0.5, 1]$ in $SEOP_{ap}$ versus the $\hat{\alpha}$ computed with Eq. (21) in Figure 1. All the results are averaged over 10 runs and the standard deviation of the observation is represented as an uncertain area. We compute the results for $|\mathcal{I}| = 10$ and $|\mathcal{M}| = 40$ since we are able to get the optimal solution in a reasonable amount of time. Moreover, with $|\mathcal{M}| = 40$ there are enough candidates to apply results in (20)–(21).

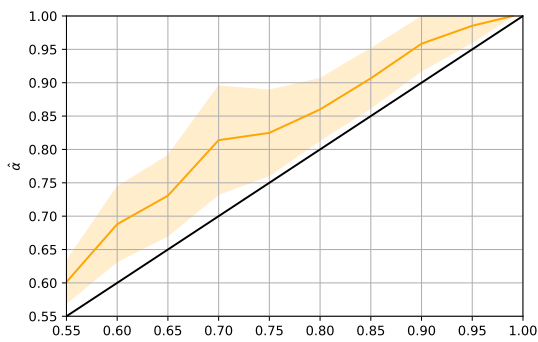


Fig. 1. Values of α set in the model vs real value of α .

As we expected, the curve is above the line $\hat{\alpha} = \alpha$ since by using the Markov inequality we are considering an upper bound on the probability. Nevertheless, the results are close to the exact value being on average 10% higher than the α set in the model. Thus, in the real field, the decision-maker

may lower by 10% the values of the α s and get a solution compliant with the wanted probability of execution.

V. CONCLUSIONS AND FUTURE WORKS

We proposed a new probabilistic model for SE-based services optimization encompassing the wta of the candidate involved in the business model. We prove, by means of CPU experiment that, despite the difficult formulation, the model can be approximated into a nice tractable form able to provide timely solution for crowd-shipping applications. However, being the SE a very seminal topic within the optimization field, we believe that a full-fledged experimental design to explore all the solution characteristics is needed. Some questions to answer are related to the performance of the method in the case in which non-concave distributions for the wta are considered or how the solutions of the model are related to the number of breakpoints used by the piece-wise wta approximation.

ACKNOWLEDGEMENTS

The work has been supported by *ULTRAOPTIMAL - Urban Logistics and sustainable TRAnspOrtation: OPTimization under uncertainTY and MACHine Learning*, a PRIN2020 project funded by the Italian University and Research Ministry (grant n. 20207C8T9M, website: <https://ultraoptimal.unibg.it>).

REFERENCES

- [1] F. Corno, L. D. Russis, and T. Montanaro, "Estimate user meaningful places through low-energy mobile sensing," in *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, Oct. 2016. doi: 10.1109/smc.2016.7844703
- [2] A. Santini, A. Viana, X. Klimentova, and J. P. Pedroso, "The probabilistic travelling salesman problem with crowdsourcing," *Computers & Operations Research*, vol. 142, p. 105722, Jun. 2022. doi: 10.1016/j.cor.2022.105722
- [3] A. Alnagar, F. Gzara, and J. H. Bookbinder, "Crowdsourced delivery: A review of platforms and academic literature," *Omega*, vol. 98, p. 102139, Jan. 2021. doi: 10.1016/j.omega.2019.102139
- [4] B. Guo, D. Zhang, Z. Wang, Z. Yu, and X. Zhou, "Opportunistic IoT: Exploring the harmonious interaction between human and the internet of things," *Journal of Network and Computer Applications*, vol. 36, no. 6, pp. 1531–1539, Nov. 2013. doi: 10.1016/j.jnca.2012.12.028
- [5] E. Fadda, G. Perboli, and R. Tadei, "A progressive hedging method for the optimization of social engagement and opportunistic IoT problems," *European Journal of Operational Research*, vol. 277, no. 2, pp. 643–652, Sep. 2019. doi: 10.1016/j.ejor.2019.02.052
- [6] —, "Customized multi-period stochastic assignment problem for social engagement and opportunistic IoT," *Computers & Operations Research*, vol. 93, pp. 41–50, May 2018. doi: 10.1016/j.cor.2018.01.010
- [7] E. Fadda, D. Mana, G. Perboli, and R. Tadei, "Multi period assignment problem for social engagement and opportunistic IoT," in *2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC)*. IEEE, Jul. 2017. doi: 10.1109/compsac.2017.173
- [8] W. M. Hanemann, "Willingness to pay and willingness to accept: how much can they differ?" *The American Economic Review*, vol. 81, no. 3, pp. 635–647, 1991.
- [9] P. Li, H. Arellano-Garcia, and G. Wozny, "Chance constrained programming approach to process optimization under uncertainty," *Computers & chemical engineering*, vol. 32, no. 1-2, pp. 25–45, 2008.
- [10] G. A. Godfrey and W. B. Powell, "An adaptive, distribution-free algorithm for the newsvendor problem with censored demands, with applications to inventory and distribution," *Management Science*, vol. 47, no. 8, pp. 1101–1112, Aug. 2001. doi: 10.1287/mnsc.47.8.1101.10231
- [11] A. Cuzzocrea, E. Fadda, and A. Baldo, "Lyapunov central limit theorem: Theoretical properties and applications in big-data-populated smart city settings," in *2021 5th International Conference on Cloud and Big Data Computing (ICCBDC)*. ACM, Aug. 2021. doi: 10.1145/3481646.3481652

Independent Component Analysis Based on Jacobi Iterative Framework and L1-norm Criterion

Adam Borowicz

Faculty of Computer Science, Bialystok University of Technology
Wiejska str. 45A, 15-351 Bialystok, Poland
Email: a.borowicz@pb.edu.pl

Abstract—Most recently, a link between principal component analysis (PCA) based on L1-norm and independent component analysis (ICA) has been discovered. It was shown that the ICA can actually be performed by L1-PCA under the whitening assumption, inheriting the improved robustness to outliers. In this paper, a novel ICA algorithm based on Jacobi iterative framework is proposed that utilizes the non-differentiable L1-norm criterion as an objective function. We show that such function can be optimized by sequentially applying Jacobi rotations to the whitened data, wherein optimal rotation angles are found using an exhaustive search method. The experiments show that the proposed method provides a superior convergence as compared to FastICA variants. It also outperforms existing methods in terms of source extraction performance for Laplacian distributed sources. Although the proposed approach exploits the exhaustive search method, it offers a lower computational complexity than that of the optimal L1-PCA algorithm.

I. INTRODUCTION

INDEPENDENT component analysis (ICA) [1] is one of the most widely used techniques in multivariate signal processing. The major goal of the ICA is to transform observed mixtures to components that are as independent from each other as possible. Since the only assumption about the components is that they are mutually independent, the ICA can be viewed as a special case of blind source separation (BSS) problem [2]. Such a problem arises in a wide range of applications, including speech/image source separation, noise reduction, feature extraction, watermark detection.

Most of the ICA algorithms are based on the central limit theorem. Among them, FastICA approach [3], [4], [5] is probably the most well-known example. It attempts to find directions in multidimensional space in which some measure of non-Gaussianity is maximized, thereby enforcing mutual independence between components. The projections of the observed multivariate data onto these directions are viewed as independent components, and often reveal much of the data's structure.

The ICA could also be seen as a generalization of the classical principal component analysis (PCA) [6] by assuming independent and non-Gaussian source distributions. In more detail, the PCA only tries to identify orthogonal directions, along which the data exhibit the greatest variability. Traditionally, this variability is measured using the Frobenius

norm (L2-norm on matrices), which allows to decorrelate the components but not to make them independent. Though, the PCA is often used in many ICA algorithms as a pre-processing step for whitening or sphering the data.

In recent years, a growing interest in approaches to the PCA based on the L1-norm can be observed [7], [8], [9]. Unlike conventional PCA, the L1-norm techniques offer an improved robustness to outliers, i.e., data points that differ significantly from the other observations. Most recently, it was shown in [10] that the ICA can actually be performed by L1-norm PCA under the whitening assumption. When the source distribution fulfills certain conditions, it is possible to extract independent components using optimal L1-PCA algorithms with guaranteed global convergence. It was demonstrated that such algorithms may give better accuracy and robustness than those of conventional ICA methods. Unfortunately, optimal L1-PCA algorithms are computationally expensive. In addition, the global convergence is guaranteed only for distributions with negative kurtosis sign. In the work [10], a new variant of the FastICA algorithm with absolute value nonlinearity was considered. Although the accuracy and robustness of this approach were comparable to that of the optimal L1-PCA algorithm, it shows serious convergence difficulties.

In this paper, a novel approach to ICA based on direct optimization of the L1-norm criterion is proposed. The method follows Jacobi iterative framework, whereby the global solution is reached by successively applying Jacobi/Givens rotations to whitened observation vectors [11], [1], [12], [13]. In this way high-dimensional ICA problem is reduced to solving a set of the simpler one-dimensional subproblems. Namely, at every iteration step, we are looking for the angle that maximizes negentropy of the transformed components. Unlike conventional Jacobi methods, the proposed algorithm exploits the contrast function based on the L1-norm, inheriting increased robustness to outliers. Since the local cost functions are of simple form, the optimal rotation angle is found using an exhaustive search method. Therefore, the differentiability of the objective function is no longer required and the method can deal with saddle points and multiple extrema. The simulation results show that the proposed method offers a superior convergence compared to the FastICA method with absolute value function nonlinearity. Furthermore, it outperforms conventional methods in source extraction performance for the mixtures with Laplacian distributions.

This work was supported by Bialystok University of Technology under the grant WZ/W1-IIT/4/2020

The paper is organized as follows. Section II describes the connection between the ICA and L1-norm criterion. It also introduces the mathematical formulations behind the FastICA technique and its recent variant using absolute value function as non-linearity. In Section III, a novel ICA algorithm is proposed based on the Jacobi iterative framework and non-differentiable L1-norm criterion. Section IV investigates the performance of the presented method via numerical simulations. Finally, the conclusions are given in Section V.

II. LINK BETWEEN ICA AND L1-PCA

Let us denote by $\mathbf{x} = [x_1, x_2, \dots, x_N]^T \in \mathbb{R}^N$ observable, zero-mean N -dimensional random vector and by $\mathbf{y} = [y_1, y_2, \dots, y_N]^T \in \mathbb{R}^N$ its linear transform, i.e. $\mathbf{y} = \mathbf{B}\mathbf{x}$. Then, the ICA problem consists of finding an unmixing matrix $\mathbf{B} \in \mathbb{R}^{N \times N}$ such that the components of \mathbf{y} are as independent as possible. Since statistical independence implies uncorrelatedness, many ICA algorithms assume explicitly that $E\{\mathbf{y}\mathbf{y}^T\} = \mathbf{I}$, where $E\{\cdot\}$ stands for expectation operator. This is usually enforced by the following factorization of the unmixing matrix:

$$\mathbf{B} = \mathbf{W}\mathbf{C}_{\mathbf{xx}}^{-1/2}, \quad (1)$$

where $\mathbf{C}_{\mathbf{xx}}^{-1/2}$ denotes the whitening matrix which is computed by inverting a square root of the observation signal covariance matrix. It can be easily verified that the constraint $E\{\mathbf{y}\mathbf{y}^T\} = \mathbf{I}$ holds for any orthogonal matrix \mathbf{W} . Since the whitening transformation is always possible, the ICA problem can be viewed as finding an orthogonal transformation of the whitened data vector $\mathbf{z} = \mathbf{C}_{\mathbf{xx}}^{-1/2}\mathbf{x}$, i.e. $\mathbf{y} = \mathbf{W}\mathbf{z}$, such that the components of \mathbf{y} are as independent as possible. This can in particular be achieved by maximizing the negentropy of the random vector \mathbf{y} defined as follows:

$$J(\mathbf{y}) = h(\mathbf{v}) - h(\mathbf{y}), \quad (2)$$

where $h(\cdot)$ is the differential entropy [14] and \mathbf{v} is the Gaussian random variable of the same covariance matrix as \mathbf{y} . Unfortunately, computation of (2) is not an easy task and in practice some approximations of negentropy [3], [5] have to be used. Let $y = \mathbf{w}^T\mathbf{z}$ denote a random variable being a linear projection of the whitened data vector onto some direction $\mathbf{w} \in \mathbb{R}^N$. Then, the negentropy of this projection can be approximated as follows:

$$J_g(y) \approx c[E\{g(y)\} - E\{g(v)\}]^2, \quad (3)$$

where c is irrelevant constant and g is any non-quadratic, sufficiently smooth even function. The variables v, y are assumed to be of zero mean and unit variance, with v being a Gaussian-distributed variable. The approximation (3) is interpreted as a measure of non-Gaussianity as it is always non-negative, and it equals to zero if and only if y is Gaussian.

In the case of the deflationary FastICA algorithm [4], the independent components are found sequentially, one after

another. For each source, the criterion (3) is optimized iteratively, using an approximate Newton technique. Namely, the following fixed-point iteration is used:

$$\hat{\mathbf{w}} = E\{\mathbf{z}g'(\mathbf{w}^T\mathbf{z})\} + E\{g''(\mathbf{w}^T\mathbf{z})\}\mathbf{w}, \quad (4)$$

$$\mathbf{w}^+ = \hat{\mathbf{w}}/\|\hat{\mathbf{w}}\|_2, \quad (5)$$

where \mathbf{w}^+ stands for the direction vector of the estimated independent component after the current iteration. These vectors are projected onto the space orthogonal to the space spanned by the earlier found vectors, so that at the end, we obtain the set $\{\mathbf{w}_i^T\}_{i=1}^N$ of orthogonal projectors that are stored in the rows of the matrix \mathbf{W} .

A. FastICA based on absolute value function

A crucial step in optimizing the FastICA algorithm is to choose the best non-linearity $g(\cdot)$ [15]. Many ICA algorithms [11], [1], [12], [16] use kurtosis-based contrast functions, which correspond to the fourth-power non-linearity $g(y) = 1/4y^4$. Such a choice can be justified on statistical grounds only for estimating sub-Gaussian sources (i.e. those with negative kurtosis) when there are no outliers. However, in practice we mostly deal with super-Gaussian variables [17] that have positive kurtosis. It was suggested in [3], [4] that for super-Gaussian densities, the optimal contrast function is a function that grows slower than quadratically. In particular, as a general-purpose contrast function, one should choose,

$$g(y) = |y|^\alpha, \quad \alpha < 2. \quad (6)$$

Nevertheless, no attempt has been made to implement this idea in practice. The reason is that the FastICA algorithm assumes the differentiability of $g(y)$, whereas for the absolute value function, this property fails at origin. Therefore, the following differentiable approximation of the absolute value function has been proposed:

$$g(y) = \frac{1}{a} \log \cosh(ay), \quad (7)$$

with $1 \leq a \leq 2$. However, this approximation may not provide the same independent source extraction performance as the absolute value function.

In the recent work [10], the authors admitted differentiability of $g(y) = |y|$ by assuming that $g'(y) = \text{sign}(y)$ and $g''(y) = 2\delta(y)$, where $\delta(y)$ denotes Dirac's delta function. It resulted in the modified FastICA method with the following iteration step in place of (4):

$$\hat{\mathbf{w}} = E\{\mathbf{z} \text{sign}(\mathbf{w}^T\mathbf{z})\} - 2f_y(0)\mathbf{w}, \quad (8)$$

where $f_y(0)$ stands for the probability density function (PDF) of y evaluated at the origin. As proposed in [10], it can be computed through the kernel density estimate with Gaussian kernel. It was also demonstrated that, for small data sizes (e.g. $N = 2, 3$), this algorithm can provide some improvements in source extraction performance when dealing with outliers. However, the iteration (8) may present difficulties converging to the right solution. Please note that, at each iteration, the PDF of the extractor output must be estimated, which may be impractical.

B. ICA via L1-PCA

Let $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_M] \in \mathbb{R}^{N \times M}$ denotes data matrix, where $\{\mathbf{z}_m\}_{m=1}^M$ are realizations of a zero-mean random vector \mathbf{z} . Assuming ergodicity conditions and that $g(y) = |y|$, it can be shown that for large enough sample size the L1-norm of the projection $\mathbf{w}^T \mathbf{Z}$ becomes proportional to $E\{g(y)\}$,

$$\|\mathbf{w}^T \mathbf{Z}\|_1 = \sum_{m=1}^M |\mathbf{w}^T \mathbf{z}_m| \rightarrow ME\{g(y)\}. \quad (9)$$

Please note that the second term in (3) is always constant. Thus, the solution is reached at a certain optimum (i.e. maximum or minimum) of $E\{g(y)\}$ under the constraint $\|\mathbf{w}\|_2 = 1$. For this reason, the ICA can also be accomplished by whitening, followed by the minimization or maximization of the L1-norm. It was shown in [10] that symmetric sources with negative (respectively, positive) kurtosis are maximizers (respectively, minimizers) of $E\{|y|\}$. Whereas, the optimization problem of finding L1 principal component can be formulated as follows [8], [9]:

$$\mathbf{w}_{L1} = \underset{\mathbf{w} \in \mathbb{R}^N, \|\mathbf{w}\|_2=1}{\operatorname{argmax}} \|\mathbf{w}^T \mathbf{Z}\|_1. \quad (10)$$

Since the objective function is non-differentiable, the problem is difficult to solve by means of conventional optimization techniques such as gradient-based methods. However, it was shown in [8] that $\mathbf{w}_{L1} = \mathbf{Z}\mathbf{c}_{\text{opt}}/\|\mathbf{Z}\mathbf{c}_{\text{opt}}\|_2$, where

$$\mathbf{c}_{\text{opt}} = \underset{\mathbf{c} \in \{\pm 1\}^M}{\operatorname{argmax}} \|\mathbf{Z}\mathbf{c}\|_2. \quad (11)$$

Hence, the L1-norm maximization can be viewed as a combinatorial problem over the binary field. A globally convergent L1-PCA algorithm with complexity $\mathcal{O}(M^{\operatorname{rank}(\mathbf{Z})})$ was proposed in [8]. A faster, yet suboptimal version of this approach [9] is based on consecutive bit-flipping operations. Though, its time complexity can still be prohibitive for large data sizes. The most computationally efficient L1-PCA method was proposed earlier in [7]. It is based on the fixed-point iterative scheme similar to that used in FastICA algorithm. Unfortunately, the method often gets trapped in local extrema. Despite these shortcomings, the L1-PCA algorithms can be used directly to extract independent sources with negative kurtosis sign (i.e. sub-Gaussians) under whitening assumption. It was shown in [10] that globally convergent L1-PCA algorithm may give better accuracy and robustness than those of the conventional ICA methods, especially when dealing with outliers. The L1-PCA algorithm can also be modified to perform L1-norm minimization. However, in such case the global convergence property is lost because the L1-norm and the L2-norm minimization problems are not related as in (10)-(11). In addition, computational complexity of this algorithm can become prohibitive for large sample size and/or observation dimensions. In most applications, only suboptimal L1-PCA algorithms [9], [7] can be considered.

III. PROPOSED METHOD

The proposed method is based on the Jacobi iterative framework [12]. Namely, an objective function is optimized by applying successively orthogonal transformations to the whitened observation data vectors:

$$\mathbf{y}^{(k+1)} = \mathbf{G}(p_k, q_k, \theta_k) \mathbf{y}^{(k)}, \quad k = 1, 2, \dots, \quad (12)$$

where $\mathbf{y}^{(1)} = \mathbf{z} = \mathbf{C}_{\mathbf{xx}}^{-1/2} \mathbf{x}$. The matrix $\mathbf{G}(p, q, \theta)$ represents Jacobi rotation [18] by the angle θ in the plane determined by the p and q coordinates, i.e.:

$$\mathbf{G}(p, q, \theta) = \begin{bmatrix} \mathbf{I}_{p-1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \cos \theta & \mathbf{0} & \sin \theta & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_{q-p-1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\sin \theta & \mathbf{0} & \cos \theta & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_{N-q-1} \end{bmatrix}, \quad (13)$$

with $1 \leq p < q \leq N$. Thus, the unmixing matrix after the k th iteration can be expressed as follows:

$$\hat{\mathbf{B}}^{(k)} = \left(\prod_{i=1}^k \mathbf{G}(p_i, q_i, \theta_i) \right) \mathbf{C}_{\mathbf{xx}}^{-1/2}. \quad (14)$$

Please note that for N -dimensional space we have $N(N-1)/2$ possible rotation planes, each uniquely represented by pair (p, q) . A sequence of rotations represented by these pairs is arranged in a so-called sweep. In fact, any rotation order is allowed, but some may work better than others [19], [20]. In this work, a typical row-cycling ordering is used as described in Tab. I. Usually, it is necessary to go through several sweeps before convergence is achieved. The algorithm is terminated when, for all rotations in the current sweep, we have $|\theta_k| < \theta_{\min}$, or when the maximum number of sweeps is reached. The parameter θ_{\min} is an empirically chosen small angle [12], which controls the accuracy of the optimization.

It is crucial for this estimation framework to compute the rotation angles θ_k so that a given objective function is gradually optimized. Motivated by the ideas presented in the previous section, we propose to maximize negentropy approximation (3) with $g(y) = |y|$ directly. Since we deal with a sequence of two-dimensional ICA problems, the objective function for two units must be considered. As suggested in [4], such a function can be defined as the sum of the one-unit functions:

$$\mathcal{J}^{(k)}(\theta) = \sum_{i \in \{p_k, q_k\}} |E\{|\hat{y}_i^{(k)}(\theta)|\} - E\{|v|\}|, \quad (15)$$

where

$$\hat{y}_{p_k}^{(k)}(\theta) = y_{p_k}^{(k)} \cos \theta + y_{q_k}^{(k)} \sin \theta, \quad (16)$$

TABLE I: Arrangement of rotation planes using a row-cycling ordering for $N = 3$.

sweep no.	1			2			...
k	1	2	3	4	5	6	...
(p_k, q_k)	(1,2)	(1,3)	(2,3)	(1,2)	(1,3)	(2,3)	...

$$\hat{y}_{q_k}^{(k)}(\theta) = y_{q_k}^{(k)} \cos \theta - y_{p_k}^{(k)} \sin \theta, \quad (17)$$

are respectively the p_k th and q_k th coefficient of the currently transformed data vector $\mathbf{y}^{(k)}$. Please note that for a normally distributed random variable v with mean μ and variance σ^2 , the random variable $u = |v|$ has a folded normal distribution. The mean of the folded distribution is given by [21]:

$$\mu_u = \sigma \sqrt{2/\pi} \exp\left(-\frac{\mu^2}{2\sigma^2}\right) + \mu \operatorname{erf}\left(\frac{\mu}{\sqrt{2}\sigma}\right) \quad (18)$$

For $\mu = 0$ and $\sigma^2 = 1$, the expectation $E\{|v|\}$ in (15) reduces to the constant factor $\sqrt{2/\pi}$. Examples of the objective function (15) evaluated at 12 consecutive data rotations are presented in Fig. 1. As we see, these functions are always periodic with period $\pi/2$. Therefore, the search for the optimal angle can be restricted to the interval $[-\pi/4; \pi/4]$, i.e.:

$$\hat{\theta}_k = \operatorname{argmax}_{-\pi/4 \leq \theta < \pi/4} J^{(k)}(\theta). \quad (19)$$

In order to construct Newton-type iteration scheme, one can admit differentiability of $g(y) = |y|$ in a similar way as for the FastICA approach. In Fig. 1 we see that the objective function contains multiple local maxima and saddle points, thus even if we use a differentiable approximation for the absolute value function, it would be difficult to reach the global maximum of (15). However, in this case, each plane rotation depends on a single parameter θ_k , reducing the N -dimensional optimization problem to the sequence of the $N(N-1)/2$ one-dimensional search subproblems per sweep. Therefore, as opposed to the FastICA approach, the solution can be found using an exhaustive method in a reasonable execution time. In particular, for each data rotation, the function (15) can be evaluated at the set of D equidistant points:

$$\theta \in \{-\pi/4 + i\pi/(2D) : i = 0, 1, \dots, D-1\}. \quad (20)$$

The greater the value of D , the better the accuracy of the optimization. For even D , the set (20) always contains a zero value. Hence, in order to ensure local convergence, the parameter θ_{\min} for stop condition should be set to any value in the interval $(0; \pi/(2D))$. We have found empirically that the rotation angles tend to decrease in subsequent sweeps, and some optimizations are possible. For example, it may not be necessary to search for optimal angle over entire interval at the later sweeps. The exhaustive search algorithm can also be replaced by more sophisticated non-gradient techniques such as simplex bisection method [22], particle swarm optimization [23], genetic algorithms [24], simulated annealing [25].

The Matlab implementation of the proposed approach is given in Alg. 1. The expectations in (15), are replaced by sums with observables in place of random variables. Please note that since the matrix $\mathbf{G}(p, q, \theta)$ modifies only (p, q) rows, it is not necessary to compute it explicitly. It is easy to see that in each sweep, we must perform $N(N-1)/2$ data rotations. Each rotation costs $4M$ multiplications, but this operation must be repeated D times as the objective function is evaluated at D points. Thus, the time complexity of the single sweep can be roughly estimated as of order $\mathcal{O}(N^2MD)$.

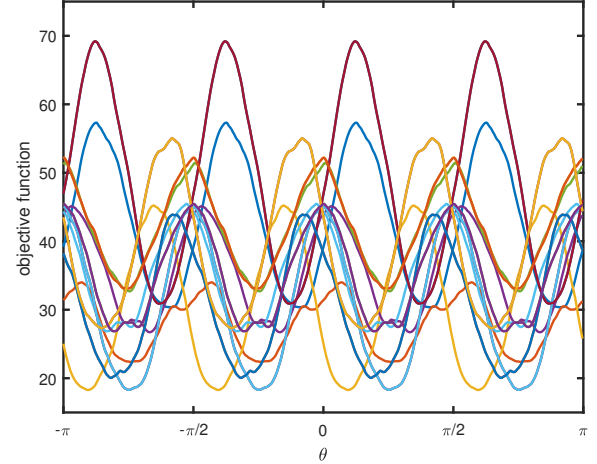


Fig. 1: Examples of the function (15) evaluated at 12 consecutive data rotations for a randomly generated mixture of $N = 4$ Laplacian distributed sources with sample size $M = 400$.

Algorithm 1 Matlab implementation of the proposed method

```

function [Y,B,k] = JICA_abs(X, k_max, D)
[N, M] = size(X);
X = X - mean(X, 2);
B = inv(sqrtm((X*X')/M));
Y = B*X;
D = D + mod(D, 2);
theta = linspace(-pi/4, pi/4-pi/2/D, D);
c = cos(theta)';
s = sin(theta)';
Gp = [ c s ];
Gq = [ -s c ];
mu = M * sqrt(2/pi);
k = 1; encore = 1;
while k <= k_max && encore
    encore = 0;
    for p = 1:N-1
        for q = p+1:N
            r = [p q];
            Yp = Gp*Y(r,:);
            Yq = Gq*Y(r,:);
            J = abs(sum(abs(Yp), 2)-mu) + ...
                abs(sum(abs(Yq), 2)-mu);
            [~, I] = max(J);
            if abs(theta(I)) > pi/4/D
                encore = 1;
                Y(r,:)=[Yp(I,:);Yq(I,:)];
                B(r,:)=[Gp(I,:);Gq(I,:)]*B(r,:);
            end
        end
    end
    k = k + 1;
end

```

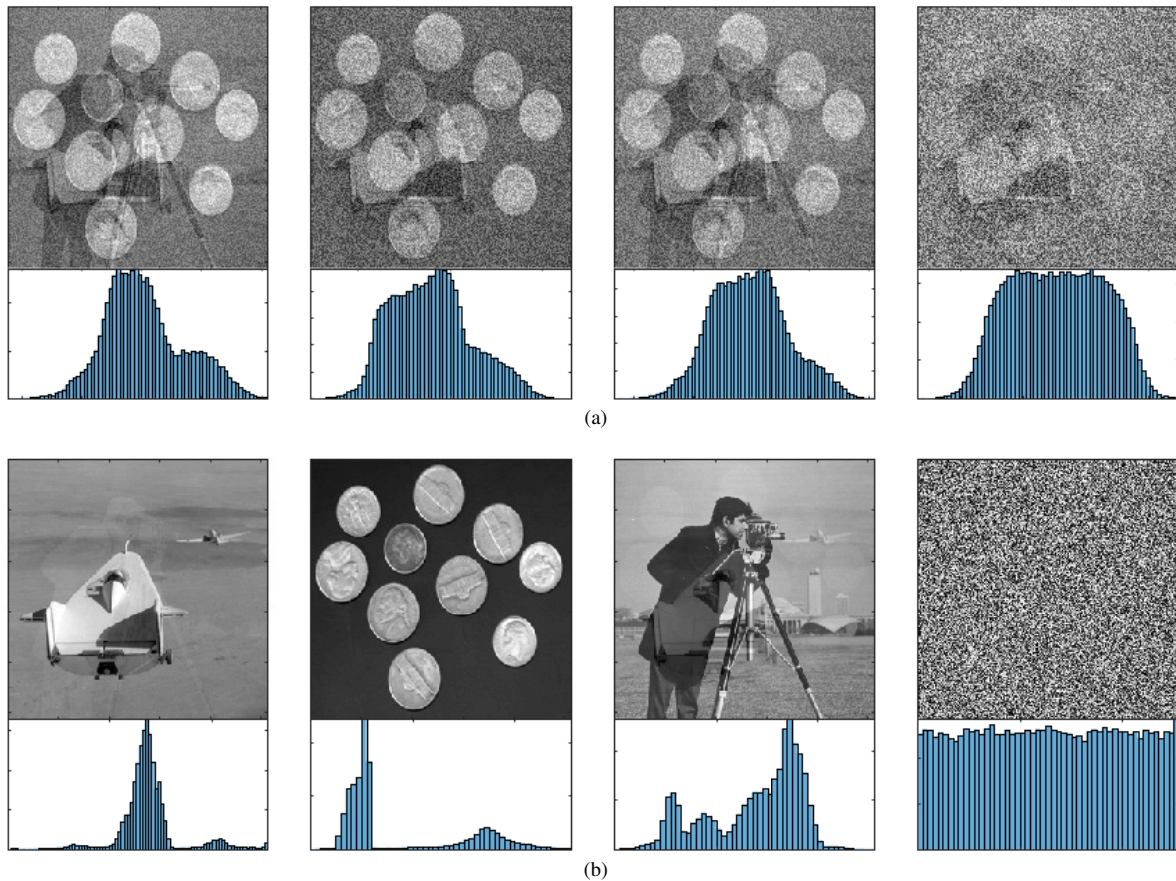


Fig. 2: Separation of the images with various distributions of pixel gray levels. (a) Randomly mixed images and histograms. (b) Recovered images and histograms.

IV. EXPERIMENTS

A. Illustrative examples

As a toy example, we considered a mixture of three Matlab built-in images and uniform noise. The images were resized to the same size 256×256 pixels with an 8-bit grayscale. The mixtures and their histograms are depicted in Fig. 2a. As can be seen, the source images have been significantly degraded, but also the form of the mixed data histograms is more like the Gaussian function. Fig. 2b shows the images recovered using the proposed method and the corresponding histograms. This example clearly shows that the algorithm is capable of transforming data from normality to independent marginal distributions.

B. Independent source extraction performance

In this experiment, the source extraction performance of the proposed algorithm is evaluated. In order to distinguish the algorithm from the existing techniques, it was denoted by the acronym JICA-abs, which stands for “Jacobi-type ICA based on absolute value function.” Also, several existing techniques including state-of-art methods were chosen for comparison, namely: joint approximate diagonalization of eigenmatrices (JADE) [11], the conventional FastICA algorithms with the

fourth-power non-linearity (FastICA-4power) and the differentiable approximation of the absolute value (FastICA-logcosh) [4], the modified FastICA algorithm based on direct use of the absolute value criterion (FastICA-abs) [10], the iterative L1-PCA [7] and more accurate bit-flipping L1-PCA method (L1-BF) [9]. In this comparison, we do not consider the optimal L1-PCA algorithm [8] due to high computational demands. On the other hand, it was shown in [10] that for the sources with uniform densities, the source extraction performance of the iterative L1-PCA method is similar to that of the optimal algorithm.

It is rather common that the performance of an iterative algorithm may vary depending on the stop conditions and initialization. Therefore, in all methods, the matrix \mathbf{W} was initialized to the identity matrix. The FastICA and iterative L1-PCA algorithms were stopped when for all sources the following condition was met: $1 - |\mathbf{w}^T \mathbf{w}^+| < \epsilon$, or when a maximum number of 1000 iterations was reached. For all these methods, except FastICA-abs, the ϵ parameter was set to 10^{-4} . In the case of the FastICA-abs, it was necessary to increase its value to 10^{-3} due to convergence difficulties. For the JICA-abs and JADE methods, the reliable and stable results were obtained when the maximum number of sweeps was set to 20

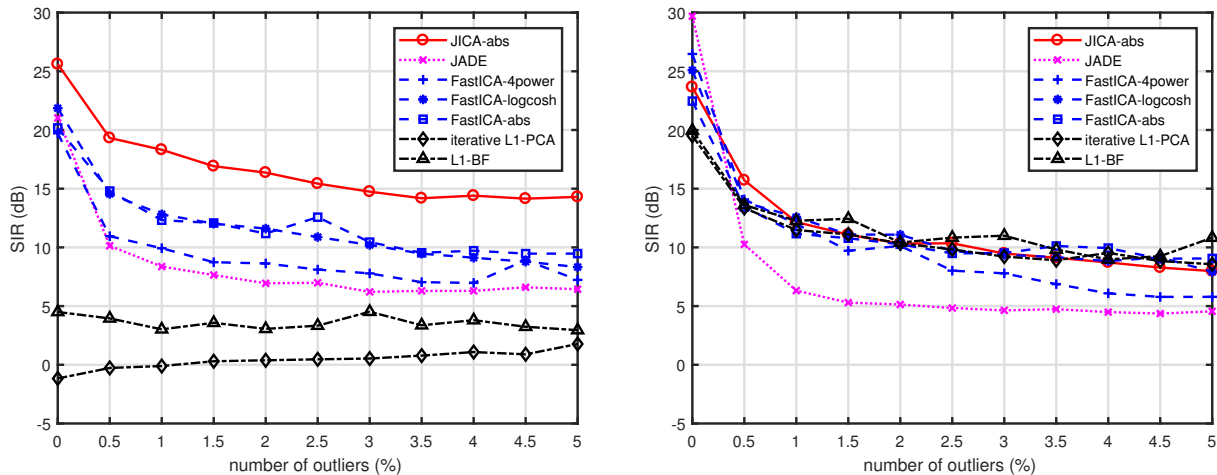


Fig. 3: Independent source extraction performance as a function of the outlier contamination rate, for $N = 4$ sources with Laplacian distribution (left), and uniform distribution (right). The length of source signals is fixed at $M = 400$ samples.

and the angle $\theta_{\min} = \pi/(4D)$ with $D = 128$. We verified empirically that rotations at a smaller angle than $\pi/512$ are not statistically significant.

For comparative purposes, we considered the linear mixtures of $N = 4$ synthetic sources, all with negative or positive kurtosis sign generated from uniform and Laplacian distribution, respectively. The length of each source signal was set to $M = 400$ samples. The coefficients of the mixing matrix were generated from the uniform distribution. Ill-conditioned matrices (with a condition number greater than 100) were excluded from the evaluation. In order to evaluate the robustness of the algorithms against outliers, randomly chosen observations were replaced with noise spikes drawn from a Gaussian distribution $\mathcal{N}(10, 1)$ at varying contamination rates.

The independent source extraction performance was estimated using the average signal-to-interference ratio (SIR) [26], [27], [28]. Please note that the higher the value of the SIR is, the better performance we get. The performance indexes were averaged over 1000 random realizations of the sources and the mixing matrices, but at each Monte Carlo run, all methods were operating on the same data. Fig. 3 presents the source extraction performance of different algorithms for various percentages of outliers. As can be seen, the proposed method clearly outperforms existing algorithms on average by 5dB for Laplacian distributed sources. In this case, the iterative L1-PCA algorithm provides the worst performance, as it is designed to only maximize L1-norm, whereas for distributions with positive kurtosis sign the L1-norm should be minimized. For Laplacian distributed sources, the L1-BF algorithm was modified to minimize the L1-norm according to the suggestion in [10]. In result, the source extraction performance of this method is slightly better than that of the iterative L1-PCA algorithm. Though, it is still poor compared to the ICA approaches. This confirms our earlier remark that the L1-norm and L2-norm minimization problems are not related in the same way as the corresponding maximization

problems (10)-(11). Although there is no clear winner for uniformly distributed sources, the JADE and FastICA-4power methods provide the best performance in the absence of outliers. Clearly, the absolute value criterion may not be the best choice for sub-Gaussian distributed sources. On the other hand, the approaches, whether based on absolute value criterion or on differentiable approximations thereof, show increased robustness to outliers as compared to the kurtosis-based methods.

C. Convergence and execution time

The total execution time of an iterative algorithm depends on the convergence rate. Unfortunately, rigorous convergence analysis of the proposed approach is not an easy task and is out of the scope of this paper. Though, we measured the average number of iterations (sweeps) taken by the presented algorithm until convergence was reached for various data sizes. The results averaged over 1000 independent runs are depicted in Fig. 4a. As can be seen that the number of iterations increases with the number of sources, but this dependency is weaker than linear, for sufficiently large M . It is rather not surprising, because as the sample size increases, an objective function usually becomes smoother and thus a faster convergence can be achieved. In this case, the algorithm converges to the stationary solution in a relatively small number of sweeps. However, please note that each sweep consists of $N(N-1)/2$ data rotations. In order to better illustrate the convergence properties of the algorithm, in Fig. 4b, we also show the global cost function measured after each data rotation for 10 independent Monte Carlo runs. It is rather clear that the algorithm converged quickly in all cases.

In order to compare the computational complexity of the proposed algorithm with the existing methods, we measured their execution times. The experiments were carried out in the Matlab environment, running on AMD Ryzen 5 3550H processor. Tab. II presents the minimum, maximum, and

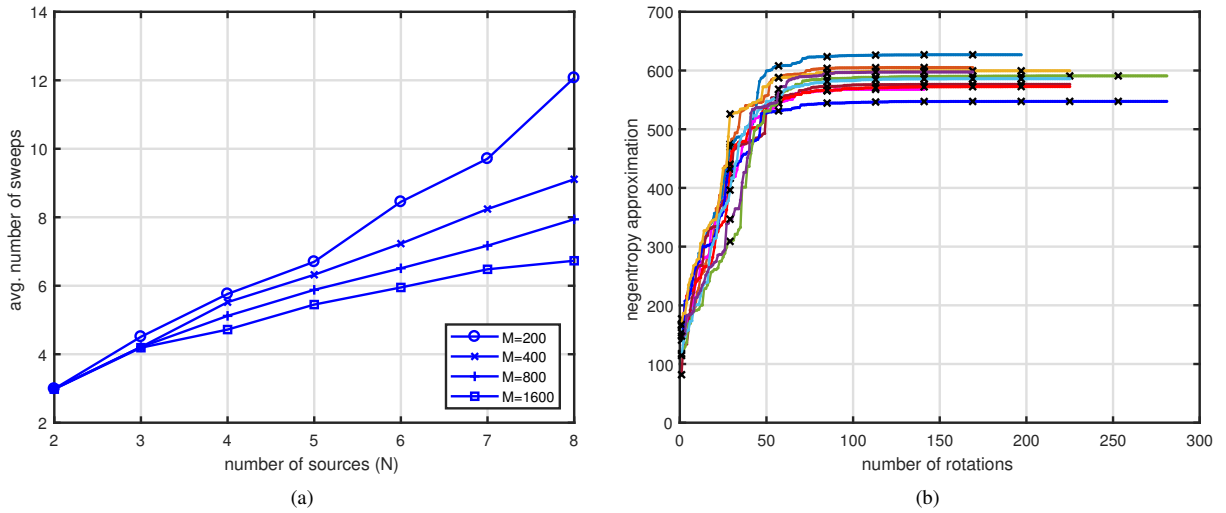


Fig. 4: Evaluation of the convergence properties of the proposed method. (a) Average number of sweeps. (b) Global contrast functions measured after each data rotation in 10 independent Monte Carlo runs. It was computed as a sum of the approximations (3) with $g(y) = |y|$ for all rows of the data matrix. The first data rotation in each sweep is denoted by cross mark.

TABLE II: Execution times (in milliseconds) and percentage of runs where the maximum number of iterations is reached in the experiments of Fig. 3 without outliers.

algorithm	Laplacian				uniform			
	t_{\min}	t_{\max}	t_{avg}	failures (%)	t_{\min}	t_{\max}	t_{avg}	failures (%)
JICA-abs	3.99	17.04	8.47	0	5.43	26.15	9.02	0
JADE	0.69	9.94	1.09	0	0.78	10.73	1.20	0
FastICA-4power	0.30	18.41	0.53	0.4	0.30	3.52	0.44	0
FastICA-logcosh	0.46	34.00	0.84	0.3	0.45	1.82	0.69	0
FastICA-abs	0.60	120.26	38.98	62.3	0.53	86.90	6.66	9.5
iter. L1-PCA	0.21	2.94	0.42	0	0.19	2.84	0.37	0
L1-BF	23.36	37.23	28.62	0	27.82	72.35	42.94	0

TABLE III: Execution times (in milliseconds) and percentage of runs where the maximum number of iterations is reached in the experiments of Fig. 3 with 5 percent of outliers.

algorithm	Laplacian				uniform			
	t_{\min}	t_{\max}	t_{avg}	failures (%)	t_{\min}	t_{\max}	t_{avg}	failures (%)
JICA-abs	3.72	18.44	8.88	0	5.37	25.12	9.07	0
JADE	0.70	10.88	1.24	0	0.68	10.54	0.99	0
FastICA-4power	0.30	16.22	0.55	0.1	0.30	18.32	1.52	5.7
FastICA-logcosh	0.47	33.86	1.16	0.3	0.51	63.12	4.51	9.4
FastICA-abs	0.52	111.34	26.01	59.8	0.62	109.39	39.93	82.1
iter. L1-PCA	0.23	3.38	0.36	0	0.24	3.59	0.36	0
L1-BF	24.80	39.64	30.34	0	26.64	81.41	46.48	0

average execution times collected in the experiment of Fig. 3 for data without outliers. The columns denoted as “failures” show the percentage of the Monte Carlo runs where the maximum iteration number was reached. The same statistics are presented in Tab. III, but for data with 5 percent of outliers. Although the JICA-abs method has relatively long average execution time, it is much faster than L1-BF algorithm. The proposed method also provides better convergence properties than those of the FastICA-based algorithms. Similarly to the JADE method and approximate L1-PCA algorithms, the JICA-

abs approach always converged to a stationary solution within the iteration limit. The FastICA-4power and FastICA-logcosh methods sometimes reach the iteration limit, which results in the increased maximum execution time. It is especially evident for uniformly distributed sources with outliers, where these methods reach the iteration limit in around 6-9 percent of runs. Unfortunately, the FastICA-abs method present even more serious convergence difficulties when dealing with outliers. In this case, the iteration limit is reached in around 82 and 60 percent of runs, for uniform and Laplacian distributions,

respectively. Obviously, the upper bound of the execution time can be reduced by decreasing the iteration limit, but it can also deteriorate the accuracy of the optimization. Therefore, a method with a smaller upper bound of the execution time may be a better choice when the timeliness of the system becomes a prominent problem.

V. CONCLUSION

A novel ICA algorithm has been proposed that directly utilizes non-differentiable absolute value criterion as a contrast function for the ICA problem. The algorithm is based on Jacobi iterative framework and exhaustive search method. Experimental studies show that the proposed approach provides better accuracy and robustness to outliers than existing methods for Laplacian distributed sources. Unlike the FastICA approaches, it does not show any convergence issues. Though, it has on average relatively high execution time as compared to the state-of-art ICA methods. On the other hand, it is faster than currently most accurate suboptimal L1-PCA algorithm that also works in an exhaustive manner.

A rigorous convergence analysis of the proposed method is of great theoretical importance, thus it should be the subject of further research. We also believe that the computational complexity can potentially be reduced. In addition, future works may include developing practical applications in speech, audio and image denoising.

REFERENCES

- [1] P. Comon, "Independent component analysis, a new concept?" *Signal Process.*, vol. 36, no. 3, pp. 287–314, 1994. doi: 10.1016/0165-1684(94)90029-9
- [2] P. Comon and C. Jutten, Eds., *Handbook of Blind Source Separation. Independent Component Analysis and Applications*, 1st ed. Oxford, USA: Academic Press, inc., 2010.
- [3] A. Hyvärinen, "New approximations of differential entropy for independent component analysis and projection pursuit," in *Proceedings of the 1997 Conference on Advances in Neural Information Processing Systems 10*, 1997, pp. 273–279.
- [4] —, "Fast and robust fixed-point algorithms for independent component analysis," *IEEE Trans. Neural Netw.*, vol. 10, no. 3, pp. 626–634, 1999. doi: 10.1109/72.761722
- [5] —, "Survey on independent component analysis," *Neural Computing Surveys*, vol. 2, 07 1999.
- [6] I. T. Jolliffe, *Principal Component Analysis*. New York: Springer Verlag, 2002.
- [7] N. Kwak, "Principal component analysis based on L1-norm maximization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 9, pp. 1672–1680, 2008. doi: 10.1109/TPAMI.2008.114
- [8] P. P. Markopoulos, G. N. Karystinos, and D. A. Pados, "Optimal algorithms for L_1 -subspace signal processing," *IEEE Transactions on Signal Processing*, vol. 62, no. 19, pp. 5046–5058, 2014. doi: 10.1109/TSP.2014.2338077
- [9] P. P. Markopoulos, S. Kundu, S. Chamadia, and D. A. Pados, "Efficient L1-norm principal-component analysis via bit flipping," *IEEE Transactions on Signal Processing*, vol. 65, no. 16, pp. 4252–4264, 2017. doi: 10.1109/TSP.2017.2708023
- [10] R. Martin-Clemente and V. Zarzoso, "On the link between L1-PCA and ICA," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 3, pp. 515–528, 2017. doi: 10.1109/TPAMI.2016.2557797
- [11] J. Cardoso and A. Souloumiac, "Blind beamforming for non-Gaussian signals," *IEE Proceedings F - Radar and Signal Processing*, vol. 140, no. 6, pp. 362–370, 1993. doi: 10.1049/ip-f-2.1993.0054
- [12] J. Cardoso, "High-order contrasts for independent component analysis," *Neural Computation*, vol. 11, no. 1, pp. 157–192, 1999. doi: 10.1162/089976699300016863
- [13] E. Learned-Miller and J. Fisher, "ICA using spacings estimates of entropy," *Journal of Machine Learning Research*, vol. 4, pp. 1271–1295, Dec. 2003.
- [14] T. M. Cover and J. A. Thomas, *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. USA: Wiley-Interscience, 2006.
- [15] A. Dermoune and T. Wei, "FastICA algorithm: Five criteria for the optimal choice of the nonlinearity function," *IEEE Transactions on Signal Processing*, vol. 61, pp. 2078–2087, 04 2013. doi: 10.1109/TSP.2013.2243440
- [16] V. Zarzoso and P. Comon, "Robust independent component analysis by iterative maximization of the kurtosis contrast with algebraic optimal step size," *IEEE Transactions on Neural Networks*, vol. 21, no. 2, pp. 248–261, 2010. doi: 10.1109/TNN.2009.2035920
- [17] A. Bell and T. Sejnowski, "An information-maximization approach to blind separation and blind deconvolution," *Neural Computation*, vol. 7, no. 6, pp. 1129–1159, 1995. doi: 10.1162/neco.1995.7.6.1129
- [18] G. Golub and C. Van Loan, *Matrix Computations*. USA: Johns Hopkins University Press, 2013.
- [19] W. Ouedraogo, A. Souloumiac, and C. Jutten, "Non-negative independent component analysis algorithm based on 2D Givens rotations and a Newton optimization," in *Latent Variable Analysis and Signal Separation*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. doi: 10.1007/978-3-642-15995-4 pp. 522–529.
- [20] M. Parfieniuk, "A parallel factorization for generating orthogonal matrices," in *International Conference on Parallel Processing and Applied Mathematics (PPAM) 2019*. Bialystok, Poland: Springer, 2019. doi: 10.1007/978-3-030-43229 pp. 567–578.
- [21] M. Tsagris, C. Beneki, and H. Hassani, "On the folded normal distribution," *Mathematics*, vol. 2, no. 1, pp. 12–28, feb 2014. doi: 10.3390/math2010012
- [22] C. Samuelsson, "Comparative evaluation of the stochastic simplex bisection algorithm and the scipy.optimize module," in *Proceedings of the 2015 Federated Conference on Computer Science and Information Systems*, ser. Annals of Computer Science and Information Systems, vol. 5, 2015. doi: 10.15439/2015F47 pp. 573–578.
- [23] T. Krzeszowski and K. Wiktorowicz, "Evaluation of selected fuzzy particle swarm optimization algorithms," in *Proceedings of the 2016 Federated Conference on Computer Science and Information Systems*, ser. Annals of Computer Science and Information Systems, vol. 8, 2016. doi: 10.15439/2016F206 pp. 571–575.
- [24] K. Pytel, "Hybrid multievolutionary system to solve function optimization problems," in *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems*, ser. Annals of Computer Science and Information Systems, vol. 11, 2017. doi: 10.15439/2017F85 pp. 87–90.
- [25] A. Alihodzic, S. Delalić, and D. Gusic, "An effective integrated meta-heuristic algorithm for solving engineering problems," in *Proceedings of the 2020 Federated Conference on Computer Science and Information Systems*, ser. Annals of Computer Science and Information Systems, vol. 21, 2020. doi: 10.15439/2020KM81 pp. 207–214.
- [26] P. Tichavsky, Z. Koldovsky, and E. Oja, "Performance analysis of the FastICA algorithm and Cramér-Rao bounds for linear independent component analysis," *IEEE Transactions on Signal Processing*, vol. 54, no. 4, pp. 1189–1203, 2006. doi: 10.1109/TSP.2006.870561
- [27] Z. Koldovský, P. Tichavsky, and E. Oja, "Efficient variant of algorithm FastICA for independent component analysis attaining the Cramér-Rao lower bound," *IEEE Transactions on Neural Networks*, vol. 17, pp. 1265–77, 10 2006. doi: 10.1109/TNN.2006.875991
- [28] A. Borowicz, "Orthogonal approach to independent component analysis using quaternionic factorization," *EURASIP Journal on Advances in Signal Processing*, vol. 2020, no. 39, p. 23, September 2020. doi: 10.1186/s13634-020-00697-0

The electric vehicle shortest path problem with time windows and prize collection

Antonio Cassia, Ola Jabali, Federico Malucelli
 Politecnico di Milano, Italy
 Email: {antonio.cassia, ola.jabali,
 federico.malucelli}@polimi.it

Marta Pascoal
 University of Coimbra, CMUC, INESC-Coimbra, Portugal,
 Politecnico di Milano, Italy
 Email: marta.brazpascoal@polimi.it

Abstract—The Electric Vehicle Shortest Path Problem (EVSP) aims at finding the shortest path for an electric vehicle (EV) from a given origin to a given destination. During long trips, the limited autonomy of the EV may imply several stops for recharging its battery. We consider combining such stops with visiting points of interest near charging stations (CSs). Specifically, we address a version of the EVSP in which the charging decisions are harmonized with the driver’s preferences. The goal is to maximize the total gained score (assigned by the driver to the CSs), while respecting the time windows and the EV autonomy constraints. We define the problem as a MILP and develop an A* search heuristic to solve it. We evaluate the method by means of extensive computational experiments on realistic instances.

I. INTRODUCTION

DEVisING tools that facilitate the use of Electric Vehicles (EVs) is key to successfully electrifying transport, e.g., [5]. The limited autonomy of EVs coupled with the scarcity of charging stations (CSs) entails that long trips may involve several charging stops. Given that such stops are likely to be time consuming, we explore the idea of matching the charging stops with user preferences.

Considering an EV that needs to travel between an origin and a destination, we consider that the user attributes a score to each CS. Such scores are based on the vicinity to Points of Interest (POIs) to the CS. For example, a user may prefer to stop at cultural venues. In this case, a high score is given to CSs nearby such venues. Using those scores, we consider the problem of optimizing the shortest path respecting all energy feasibility constraints, while maximizing the total score that the user can achieve. To handle this problem we propose the Maximum Profit Model (MPM). To avoid excessively long trips, we limit the duration of the route. We establish this limit by solving the Shortest Path Model (SPM), which is the shortest EV path in time. We allow the MPM to deviated from the shortest path length within a given tolerance.

We adapt the Mixed Integer Linear programming (MILP) model proposed in [6] to handle the MPM and the SPM. Furthermore, we develop a heuristic algorithm based on the A* search which handles large instances of MPM. To this end, we propose the Maximum Discounted Profit Model (MDPM),

This work was partially supported by the Portuguese Foundation for Science and Technology (FCT) under project grants UID/MAT/00324/2020 and UID/MULTI/00308/2020.

which discounts the scores of the nodes into the arc costs that connect them. By doing so, we are able to efficiently adapt the A* search Algorithm to the MPM.

In general, the Electric Vehicle Shortest Path Problem (EVSP) is a shortest path problem that accounts for battery limitation and charging constraints. Due to their limited autonomy, EVs may need to detour to CSs in order to recharge their battery. This is particularly true in medium and long range routes, like in [11]. A key decision in this context is where and how much to charge the EVs. The problem of minimizing the overall trip time for EVs in road networks was studied by [1]. A heuristic algorithm for solving large EVSP instances was proposed in [14]. Baum et al. [2] introduced a functional representation of the optimal energy consumption between two locations, which led to developing an efficient heuristic algorithm that computes energy optimal paths.

One of the main EVSP modeling assumption relate to how the EV batteries are recharged. Some researchers assume that the EV must completely recharge before leaving a CS, e.g. [4], [9]. Other works (e.g., [6], [10], [12]) consider the charging quantity as a decision within the optimization. As in most studies we assume that the energy consumption is directly and exclusively related to the traveled distance.

In practice, the EV charging function is nonlinear with respect to time, and depends on the used charging technology. The nonlinear charging functions were modeled as piecewise linear concave functions by [6], [10].

Time window (TW) constraints have been introduced in EV problems by [12]. TWs oblige EVs to arrive in predetermined CSs before or during a particular time interval. Contrary to what is done in the literature, we assume that TW types are given (e.g., lunch breaks), yet their location is determined from a set of CSs which accommodate this type of TW. Furthermore, we consider required and weakly mandatory TWs.

Considering that CSs have different scores and a given maximum path length (in time), the aim of the MPM is to maximize the total score of visited CSs. As such, CSs that better match the user preferences are prioritized. We consider a single EV, with partial recharging decisions and nonlinear charging functions. Furthermore, we consider only public CSs, having different technologies and scores. We also consider TWs with a limit on the total travel time.

The contributions of this paper are threefold: 1) we propose an EV shortest path model that accounts for user preferences (**MPM**); 2) we develop a heuristic based on the A* algorithm to solve that problem; 3) and we verify the performance of both the MILP model and the heuristic algorithm on realistic test instances.

We introduce the **MPM** in Sec. II and develop an A* algorithm for it in Sec. III. We present our computational experiments in Sec. IV, and state our conclusions in Sec. V.

II. PROBLEM DEFINITION

The **MPM** maximizes the score of the CSs visited by the EV, such that the resulting path is feasible and within a tolerance from the shortest EV path. In the literature, the goal of maximizing scores obtained by visiting nodes is often called prize collection [13]. To model the **MPM** we adapt the arc based MILP model of [6]. Due to space limitations, we do not present the full formulation. Instead, we give an overview of that work and then detail the major adaptations made to handle the **MPM**.

Froger et al. [6] introduced the Fixed Route Vehicle Charging Problem (FRVCP). Given a sequence of customer nodes (i.e., not CSs) to visit, the objective of the FRVCP is to determine the charging operations (which CSs to visit and how much to charge), in order to minimize the total route duration while satisfying the following conditions: The customers in the resulting route are visited according to the given order, the resulting route is energy feasible and satisfies a maximum duration limit T^{\max} . The state of charge (SoC) of the EV is tracked on each traversed arc and visited node. The EV may be partially recharged at a set of charging stations \mathcal{S} , which may have different technologies. For each technology we consider a nonlinear charging function, approximated via a piecewise linear function following the models by [10]. In the **MPM** the fixed sequence of nodes to visit goes from an origin \mathcal{O} node to a destination \mathcal{D} node.

Building on the FRVCP, we now describe the **MPM**. Let $\mathcal{G} := (\mathcal{S}_{\mathcal{O},\mathcal{D}}, \mathcal{A})$ be a directed graph, where $\mathcal{S}_{\mathcal{O},\mathcal{D}} := \mathcal{S} \cup \{\mathcal{O}, \mathcal{D}\}$ and \mathcal{A} is the set of arcs that connect pairs of nodes in $\mathcal{S}_{\mathcal{O},\mathcal{D}}$. Let $t_{ij} \geq 0$ and $e_{ij} \geq 0$ be the driving time and energy consumption of arc $(i, j) \in \mathcal{A}$, both satisfying the triangular inequality. The EV departs from \mathcal{O} with a fully charged battery of capacity Q . We impose that the SoC of the EV is at least q_{\min} throughout the route. Since we consider long trip planning, we assume that the number of nights is a user input.

The user will spend some time in the neighborhood of the selected CS. Moreover, in certain moments of the day, the user may prefer to visit a CS near a POI, e.g., restaurants or hotels. CSs are typically strategically placed near those types of POIs. It is important that the user visits CSs that best suit her preferences. We attribute a score σ_j for $j \in \mathcal{S}$, which are input to the **MPM**. We assume that they can be derived based on the user's ranking of POIs. Furthermore, the user may personalize her trip by imposing TWs related to an activity (e.g., lunch breaks). The classical definition of TWs determines a time to visit a given node. In the **MPM**, such TWs may be realized

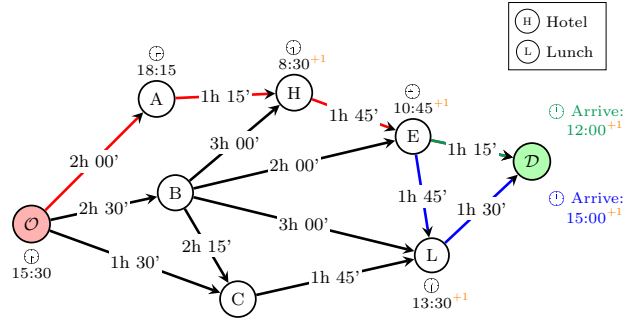


Figure 1: Path from \mathcal{O} to \mathcal{D} with a night at a hotel and lunch breaks. The timestamp near a node represents the corresponding departure time, including the recharging time (not reported in the figure). For \mathcal{D} , instead, the timestamps represent the arrival time. The number on each arc is its travel time. The red path is optimal, stopping in the hotel as required and reaching node E . If also the lunch break is required, then the EV is forced to arrive to \mathcal{D} with stopping at L (blue path). Instead, with the weakly mandatory flag, from E the EV can go directly to \mathcal{D} 3h in advance (green path).

at a number of CSs. We therefore denote the TWs related to our **MPM** as Activity-based Time Windows (ATWs). In our experiments in Sec. IV, we assign to each CS a randomly generated score between 0 and 5. In practice, these values will be established by the user in real applications.

Let \mathcal{W} be the set of all ATWs, which is partitioned into two subsets, $\mathcal{W}^R, \mathcal{W}^M \subseteq \mathcal{W}$, formed by the required and the weakly mandatory ATWs. The EV is forced to stop in each $k_R \in \mathcal{W}^R$, but it must stop in $k_M \in \mathcal{W}^M$ only if there exists at least one k_R such that $k_M \prec k_R$. Otherwise, k_M may be skipped. We assume that the sets $\mathcal{W}^R, \mathcal{W}^M$ are ordered by chronological order of starting time of the TWs. Each ATW is defined as:

$$k := (\gamma_k^L, \gamma_k^U, t_k^{\min}, o_k, \nu_k), \quad k \in \mathcal{W},$$

where the interval $[\gamma_k^L, \gamma_k^U]$ (with $\gamma_k^L < \gamma_k^U$) describes its initial and ending times (the EV must arrive before γ_k^U); t_k^{\min} is the minimum time that the EV needs to stop during k ; o_k is a binary value, equal to 1 if $k \in \mathcal{W}^M$, and 0 otherwise. The set \mathcal{W} is ordered, thus, if $k \prec h$, then $\gamma_k^L < \gamma_h^L$, for any $k, h \in \mathcal{W}$. The EV is allowed to arrive at a node with a maximum anticipation time $\tilde{\varphi}$. In this case, the activity will nevertheless start at γ_k^L . Thus, if the EV arrives in node i in the interval $[\gamma_k^L - \tilde{\varphi}, \gamma_k^U]$, then it may decide to stop at i for at least t_k^{\min} or instead perform k in a subsequent node. In addition, we use hard TWs, which means that it is not possible to perform the ATW k before $\gamma_k^L - \tilde{\varphi}$ nor after γ_k^U . Finally, two ATWs can overlap, but they cannot be contained in one another. Each CSs is associated with multiple POIs, and each TW requires a specific POI. This information is stored in the label ν_k and is different for each TW. For instance, if $k \in \mathcal{W}$ refers to the first night, then $\nu_k = \text{"Hotels"}$ and the EV is forced to stop at a CSs near a hotel. So, it is possible to construct the set of chargers $\mathcal{S}_k \subseteq \mathcal{S}$ that have in their neighborhood the POI stated in ν_k .

To determine T^{\max} we solve an EV Shortest Path Problem **SPM**, subject to the same constraints as **MPM**, but with the goal of minimizing the total trip duration (i.e., ignoring the scores). Let T^{opt} be the optimal objective function value of **SPM**, which is a theoretical lower bound on T^{\max} . Then, let T^{add} be the tolerance of the total additional detouring time from the shortest path to stop in nodes with higher scores. Therefore, we set T^{\max} to $T^{\text{opt}} + T^{\text{add}}$.

To avoid routes with many stops at SCs, we introduce the parameter r^{\min} , as $r^{\min} := \lfloor 0.4(Q - q_{\min})/\eta \rfloor$, where η is the average energy consumption per kilometer (expressed in kWh/km), and is dependent on the EV type. The ratio $(Q - q_{\min})/\eta$ represents the maximum autonomy of the vehicle, excluding the minimal amount of energy that is always required. With this arrangement it is possible to prune all the arcs associated with a distance less than r^{\min} , which also avoids stops for charging the EV in consecutive locations that are very close to each other.

Let ξ denote a lower bound on the distance of a trip from \mathcal{O} to \mathcal{D} (its computation is described in Sec. IV). Then the maximum number of legs N in a path is defined as $\lceil 1.5[\xi/r^{\min}] \rceil$.

III. A* SEARCH ALGORITHM

In the following we propose a heuristic algorithm for the **SPM** and extend it for solving the **MPM**. Recall that the **SPM** solution serves as an input to **MDPM**. The algorithm is based on the A* search, which finds a path from an origin \mathcal{O} to a destination \mathcal{D} with the smallest cost. To do that, it maintains the tree of all the paths originated from \mathcal{O} and extends each of them one arc at a time until \mathcal{D} is reached. It uses a best-first search by selecting the node that minimizes

$$f(i) := g(i) + h(i),$$

where i is the current node, $g(i)$ is the cost of the path from \mathcal{O} to i , and $h(i)$ is an estimate of the cost of the shortest path from i to \mathcal{D} . If $h(i)$ never overestimates the real cost to reach \mathcal{D} from i , for all i , then the A* algorithm finds the optimal solution.

The A* algorithm is based on minimizing cost mechanisms, thus it is not directly applicable to prize collection settings such as the **MDPM**. Therefore, we propose the **MDPM** which approximates the **MPM**, searching for a shortest path while maximizing the total score. To do that, we assign a weight \tilde{s}_{ij} with each arc, defined as

$$\tilde{s}_{ij} := t_{ij} + \Delta_j - \mu\sigma_j, \quad \forall (i, j) \in \mathcal{A} \quad \text{s.t.} \quad i, j \in \mathcal{S},$$

where Δ_j is the time spent waiting while the EV is charging at the CS j , and μ is a parameter that indicates the relative importance of the score with respect to the time required to go from i to j , charging time included. The objective function of the **MDPM** model is then

$$\min \sum_{(i,j) \in \mathcal{A}, j \in \mathcal{S}} \tilde{s}_{ij} x_{ij}.$$

This expression is nonlinear, but it can be linearized by introducing new decision variables s_{ij} , and changing the objective function as

$$\min \sum_{(i,j) \in \mathcal{A}, j \in \mathcal{S}} [(t_{ij} - \mu\sigma_j)x_{ij} + s_{ij}]$$

We first discuss the computation of the potentials used to estimate the heuristic function h . We then incorporate the TWs in the heuristic. Finally, we modify the algorithm to account for the scores in h .

Let q_i and \bar{q}_i be the SoC when the EV arrives and departs from CS i . The variables \underline{c}_i and \bar{c}_i are respectively the start and end time for charging an EV. Variables $\underline{\tau}_i, \bar{\tau}_i$ track the time when the EV arrives and leaves CS $i \in \mathcal{S}$, respectively. There is also a tolerance φ_i that represents how much time in advance, with respect to γ_k^L , the EV can arrive in i , for any $i \in \mathcal{S}_{\mathcal{D}}$. The maximum anticipation time is set to $\bar{\varphi}$, but even if the EV arrives in advance, the minimum stopping time t_k^{\min} starts at γ_k^L and not before.

Let x_{ij} be a binary variable which equals 1 if the EV arrives at node j from node i , 0 otherwise. The variable y_{jk} is also binary and it is 1 if the EV stops in j in TW k , 0 otherwise. The maximum duration of the trip is T^{\max} (in Sec. III we show how to compute an upper bound for this value). The variable z_k is binary and is equal to 1 if the EV arrives in \mathcal{D} after TW k , 0 otherwise, for any $k \in \mathcal{W}^M$. It is used to link the arrival time in node \mathcal{D} and avoidable TWs.

A. Potentials

We now find an initial estimate of the total time from any CS i to \mathcal{D} , building on techniques used in [14]. We start by dropping the charging and the TW constraints, thus we obtain a problem that can be solved using the Dijkstra's algorithm. Let $\mathcal{G} := \langle \mathcal{S}_{\mathcal{O}, \mathcal{D}}, \mathcal{A} \rangle$ be the directed graph from \mathcal{O} to \mathcal{D} , where $\mathcal{S}_{\mathcal{O}, \mathcal{D}}$ is the set of nodes and $\mathcal{A} := \mathcal{S}_{\mathcal{O}} \times \mathcal{S}_{\mathcal{D}}$ the set of arcs. We then apply the backward Dijkstra's algorithm, which operates on the reversed graph $\mathcal{G}' := \langle \mathcal{S}_{\mathcal{O}, \mathcal{D}}, \mathcal{A}' \rangle$ where

$$\mathcal{A}' := \mathcal{S}_{\mathcal{D}} \times \mathcal{S}_{\mathcal{O}} \text{ such that } (i, j) \in \mathcal{A} \Rightarrow (j, i) \in \mathcal{A}',$$

while considering \mathcal{D} as the origin and \mathcal{O} as the destination. This allows us to obtain a lower bound on the driving time from any $i \in \mathcal{S}_{\mathcal{O}, \mathcal{D}}$ to \mathcal{D} , which we denote by $\pi_{\text{dr}}(i)$. To account for energy consumption, we apply the backward Dijkstra's algorithm considering the energy consumption as the weight for the arcs. This allows us to derive the minimum amount of energy required from $i \in \mathcal{S}_{\mathcal{O}, \mathcal{D}}$ to \mathcal{D} . We denote this lower bound by $\pi_{\text{cons}}(i)$.

The EV arrives partially charged at each node, with an amount of energy equal to $\text{SoC}(i)$. Since the minimal amount of SoC q_{\min} needs to be respected at every node, we can think of $\text{SoC}(i) - q_{\min}$ as the available energy at node $i \in \mathcal{S}_{\mathcal{O}, \mathcal{D}}$. Then, to compute the minimal amount of energy from i to \mathcal{D} , we define

$$\tilde{\pi}_{\text{cons}}(i) := \pi_{\text{cons}}(i) - (\text{SoC}(i) - q_{\min}).$$

We now compute a lower bound for the charging time, based on the minimal required energy at a node $i \in \mathcal{S}_{\mathcal{O}, \mathcal{D}}$. We define $\mathcal{G}_i := \langle \mathcal{T}_i, \mathcal{A}_i \rangle$ as a subgraph of \mathcal{G} , where \mathcal{T}_i is the set of reachable nodes from i , and \mathcal{A}_i is the set of arcs comprising a path from i to any $j \in \mathcal{T}_i$ (see Fig. 2).

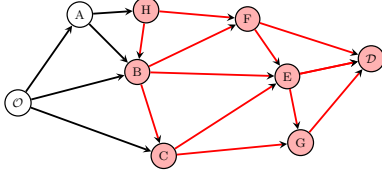


Figure 2: Illustration of \mathcal{G}_H

Let $s_{\max}(i)$ denote the maximum charging rate of all the CSs in \mathcal{T}_i . We denote the maximum charging rate of CS j as $\bar{\rho}_j$, which corresponds to the largest slope of the piecewise charging function of j . We then set $s_{\max}(i)$ as $\max\{\bar{\rho}_j : \forall j \in \mathcal{T}_i\}$. This improves the computation of the charging potential with respect to [14], where s_{\max} is constant and does not depend on the nodes that are reachable from i .

If the energy available at i is greater than the remaining energy needed to reach \mathcal{D} , then $\tilde{\pi}_{\text{cons}}$ can be negative. Thus, two cases are considered when computing a lower bound for the charging time:

$$\pi_{\text{ch}}(i) := \begin{cases} \frac{\tilde{\pi}_{\text{cons}}(i)}{s_{\max}(i)} & \text{SoC}(i) - q_{\min} \leq \pi_{\text{cons}}(i) \\ 0 & \text{otherwise} \end{cases}$$

This gives a potential that returns the minimal charging time from any node i to \mathcal{D} . Thus, a lower bound on the total trip time is given by

$$\tilde{\pi}_{\text{tt}}(i) := \pi_{\text{dr}}(i) + \pi_{\text{ch}}(i) \quad \forall i \in \mathcal{S}_{\mathcal{O}, \mathcal{D}}.$$

We improve this lower bound by accounting for TWs. Let $\pi_{\text{tw}}(i)$ be the minimal stopping time that the EV must perform from i to \mathcal{D} according to the ATWs. The trip accounts for the sum of all the minimum stopping times. Let \tilde{k} be the last TW in the ordered set \mathcal{W}^R . Suppose that, when at node i , the EV has not performed all TWs in \mathcal{W}^R . Then, $g(i) \leq \gamma_{\tilde{k}}^L + t_{\tilde{k}}^{\min}$, where $g(i)$ is the arrival time at i and $\gamma_{\tilde{k}}^L$ is the starting time of TW \tilde{k} . In this case, we compute a lower bound for the TWs potential as the sum of all the stopping times that have not been performed by the time $g(i)$. If instead, in node i , the EV has already done all the TWs in \mathcal{W}^R , then $g(i) > \gamma_{\tilde{k}}^L + t_{\tilde{k}}^{\min}$ and so the potential for the TWs must be zero. Therefore, if $\pi_{\text{tw}}(i)$ is the TWs potential for node i , we have

$$\pi_{\text{tw}}(i) := \begin{cases} \sum_{k \in \mathcal{W}^R: g(i) < \gamma_k^L} t_k^{\min} & \text{if } g(i) \leq \gamma_{\tilde{k}}^L + t_{\tilde{k}}^{\min} \\ 0 & \text{otherwise} \end{cases}$$

For instance, suppose that \mathcal{W}^R contains a 1 hour stop for lunch, one tourism stop for 2 hours and one for sleeping for

11 hours. Then, before lunch $\pi_{\text{tw}}(i) = 14$, after lunch $\pi_{\text{tw}}(i) = 13$, and the next day $\pi_{\text{tw}}(i) = 0$.

We now incorporate π_{tw} in $\tilde{\pi}_{\text{tt}}$. The EV charges during each stop. Thus, we cannot simply sum π_{tw} and π_{ch} , since they may overlap. Instead, we can obtain a better lower bound considering the maximum between π_{ch} and π_{tw} , and then adding the driving potential π_{dr} . So,

$$\pi_{\text{tt}}^1(i) := \pi_{\text{dr}}(i) + \max\{\pi_{\text{ch}}(i), \pi_{\text{tw}}(i)\}$$

defines the lower bound for the total stopping time from i to \mathcal{D} . We are not overestimating the real cost, since taking the maximum we consider for each node the best possible charging scenario that at least the EV has to perform.

B. Labels

Label L_{j_m} represents the state of the EV when arriving at the m -th copy of node j , that is dynamically allocated. With this label we keep track of the state of the EV. For notational convenience, we denote $g(j_m)$ by g_j^m . Each label includes the total time needed to reach j_m , so it includes both driving and charging times. For instance, suppose that the EV goes from the n -th copy of i to the m -th copy of j , namely from i_n to j_m . Then the label L_{j_m} considers driving from i to j and the charge in i that is needed to reach j for the m -th time. It excludes the time the EV spends charging in j . The label of the m -th copy of node j , with i_n as its direct predecessor, is:

$$L_{j_m} := (i, g_j^m, h_j^m, f_j^m, p_j^m, \beta_j^m, q_j^m, \underline{q}_j^m, \lambda_j^m, \Delta_j^m, \omega_j^m)$$

where

- i is the node from which the EV arrives to j_m ;
- g_j^m is the total travel time from \mathcal{O} to j_m ;
- h_j^m is the estimated travel remaining time from j_m to \mathcal{D} ;
- f_j^m is the estimated arrival time at \mathcal{D} if the path from \mathcal{O} to j_m is performed. It is given by $f_j^m := g_j^m + h_j^m$;
- p_j^m is the label from which the EV arrives, i.e., L_{i_n} , which is the label of the n -th copy of node i , with $n \in \{1, \dots, M_i\}$;
- β_j^m is the additional time spent at j for charging, it is chosen from an ordered set $\beta := \{\beta_1, \beta_2, \dots, \beta_s\}$;
- q_j^m is the amount of energy that is charged in the predecessor node i_n . It is computed to at least respect the consumption e_{ij} and is given by $q_j^m := \bar{q}_j^m - \underline{q}_j^m$, where

$$\bar{q}_j^m := \max\{q_i^n + e_{ij}, Q\};$$

- \underline{q}_j^m is the amount of energy of the EV when arriving at j_m . It is computed as $\underline{q}_j^m := q_i^n + q_j^m - e_{ij}$;
- λ_j^m is the minimum time the EV must charge at j_m . This amount of time is considered in the next label and not in L_{j_m} . It is defined as

$$\lambda_j^m := \begin{cases} \max\{0, \gamma_k^L - g_j^m\} + t_k^{\min} & \text{if TW } k \text{ is} \\ & \text{performed in } i_n \\ 0 & \text{otherwise} \end{cases}$$

where γ_k^L is the starting time of TW k and t_k^{\min} is its minimum stopping time. TW k is retrieved using ω_j^m (see below);

- Δ_j^m is the charging time in i_n . It is given by

$$\Delta_j^m := \max \{ \lambda_i^n, \bar{c}_j^m - \underline{c}_j^m \} + \beta_j^m,$$

where $\bar{c}_j^m := \Phi_j^{-1}(\bar{q}_j^m)$ and $\underline{c}_j^m := \Phi_j^{-1}(\underline{q}_j^m)$, with Φ_j^{-1} the inverse of the charging function in node j . The term β_j^m is added to consider the cases in which the EV charges more than needed;

- ω_j^m is the index of the last TW k prior to node j_m .

Parameters Δ , q and \underline{q} are strictly related to β . As a consequence, also g , h and f depend on β . The value of λ instead depends strictly on ω .

C. A* search algorithms for the MPM and MDPM

Using the labels described above we now outline the A* search algorithm. We start by implementing a heuristic approach to find the fastest path from \mathcal{O} to \mathcal{D} , referring to this as **AsM**.

The origin node \mathcal{O} is initialized as follows:

$$L_{\mathcal{O}}^1 := (\mathcal{O}, g_{\mathcal{O}}^1 = t_{\text{start}}, h_{\mathcal{O}}^1 = h^1(\mathcal{O}), f_{\mathcal{O}}^1 = g_{\mathcal{O}}^1 + h_{\mathcal{O}}^1, p_{\mathcal{O}}^1 = -, 0, 0, \underline{q}_{\mathcal{O}}^1 = Q, 0, 0, 0),$$

where $g_{\mathcal{O}}^1$ is the starting time of the trip. Thus, $f_{\mathcal{D}}^m$ represents the arrival time in \mathcal{D} and not the duration of the trip. Let \mathcal{L} be the set of all labels, and M_j be a counter of the number of copies of $j \in \mathcal{S}_{\mathcal{O}, \mathcal{D}}$. The algorithm keeps track of open labels using a priority queue \mathcal{Q} . Every time a new label is created, it is added to \mathcal{Q} in a way that the first element of \mathcal{Q} is always the one with the lowest f_j^m , among all labels L_j^m , so

$$\bar{j}_m := \arg \min_{j_m: j \in \mathcal{S}_{\mathcal{O}, \mathcal{D}}, m=1, \dots, M_j} \{ f_j^m \}.$$

We now introduce some of the functions used later in the pseudocode. The function $\text{POP}(\mathcal{Q})$ returns the label with the lowest f in \mathcal{Q} , while function $\text{PUSH}(\mathcal{Q}, L_{j_m})$ adds the label L_{j_m} to the queue. Function $\text{STAR}(\mathcal{G}, j)$ returns the set of nodes $h \in \mathcal{S}_{\mathcal{D}}$ such that $(j, h) \in \mathcal{A}$, in descending order with respect to t_{jh} . The function $\text{NEXTTW}(\mathcal{W}^R, \omega_j^m)$ returns the next TW in \mathcal{W}^R that is not visited when the EV arrives at node j_m . The EV is allowed to arrive at a node with a maximum anticipation time of $\bar{\varphi}$. The boolean function $\text{NODEHASPOI}(i, \nu)$ returns one if there is at least one POI of the category ν in the neighborhood of node i , and zero otherwise.

Function $\text{MAXSLOPE}(\hat{S})$ applied to a generic subset \hat{S} of CS \mathcal{S} , returns the maximum slope between all the charging functions of nodes in \hat{S} . To speed up the algorithm, all the subtrees are precomputed. The function $\text{ROUTING}(i, j)$ returns the pair (t_{ij}, e_{ij}) , that are respectively the time and the energy required to go from i to j , for any $i \in \mathcal{S}_{\mathcal{O}}, j \in \mathcal{S}_{\mathcal{D}}$. The function $\text{MINSTOP}(g_i)$ returns $\pi_{\text{tw}}(i)$.

The A* algorithm is described in Alg. 1. It starts by initializing the counters for all the copies and storing the subtree of each node. Then the label associated with the origin is created and added to the queue and to the set of all the labels.

Algorithm 1 ASTARSEARCH Algorithm

```

1: function ASTARSEARCH( $\mathcal{G}$ ,  $Q$ ,  $q_{\min}$ ,  $t_{\text{start}}$ ,  $t_{\text{end}}$ ,  $\mathcal{W}^R$ ,  $\beta$ )
2:   for all node  $h \in \mathcal{S}_{\mathcal{O}, \mathcal{D}}$  do
3:      $\mathcal{T}[h] := \text{SUBGRAPH}(\mathcal{G}, h)$ ,  $M_h := 0$ 
4:   end for
5:    $M_{\mathcal{O}} := 1$ , Initialize  $L_{\mathcal{O}}^1$ ;  $\mathcal{L} := \{L_{\mathcal{O}}^1\}$ ;  $\mathcal{Q} := \{\mathcal{O}\}$ 
6:   while  $\mathcal{Q}$  do
7:      $L_i^n := \text{POP}(\mathcal{Q}) = (\bar{i}, g_i^n, h_i^n, f_i^n, p_i^n, \beta_i^n, q_i^n, \underline{q}_i^n, \lambda_i^n, \Delta_i^n, \omega_i^n)$ 
8:      $k := \text{NEXTTW}(\mathcal{W}^R, \omega_i^n)$ 
9:     if  $i = \mathcal{D}$  then
10:      if  $\omega_i^n < |\mathcal{W}^R|$  then go to 6
11:      return  $L_i^n$ ,  $\mathcal{L}$ 
12:     end if
13:     if  $k$  is not NONE then // See A
14:        $\text{IDX} := \omega_i^n$ ;  $\text{C} := k$ 
15:       while  $\text{C}$  is not NONE do
16:         if  $\mathcal{T}[i] \cap \mathcal{S}_{\mathcal{C}} = \emptyset$  then go to 6
17:          $\text{IDX} := \text{IDX} + 1$ ;  $\text{C} := \text{NEXTTW}(\mathcal{W}^R, \text{IDX})$ 
18:       end while
19:     end if
20:      $\text{NEIGHBORS} := \text{STAR}(\mathcal{G}, i)$ 
21:     for all  $j \in \text{NEIGHBORS}$  do
22:        $t_{ij}, e_{ij} := \text{ROUTING}(i, j)$ 
23:       if  $e_{ij} > Q - q_{\min}$  then go to 21
24:       for all  $\beta \in \beta_{\mathcal{O}}$  do
25:          $\Delta, q := \text{CHARGINGENERGY}(i, e_{ij}, \underline{q}_i^n)$ 
26:          $\Delta := \max \{ \lambda_i^n, \Delta \} + \beta$ ,  $q := \text{CHARGINGFORTIME}(i, \underline{q}_i^n, \Delta)$ 
27:          $g_{\text{temp}} := g_i^n + \Delta + t_{ij}$ 
28:         if  $g_{\text{temp}} > t_{\text{end}}$  then go to 21
29:         if  $k$  is not NONE then
30:           if  $g_{\text{temp}} > \gamma_k^U$  then go to 21
31:           if  $\gamma_k^L - \bar{\varphi} \leq g_{\text{temp}} \leq \gamma_k^U$  then
32:             if  $j = \mathcal{D}$  or not  $\text{NODEHASPOI}(j, \nu_k)$  then go to 39
33:              $M_j := M_j + 1$ ;  $m := M_j$ 
34:              $\lambda := \max \{ 0, \gamma_k^U - g_{\text{temp}} \} + t_k^{\min}$ 
35:              $L_j^m := \text{CREATELABEL}(i, j, g_i^n, \underline{q}_i^n, \Delta, q, t_{ij}, e_{ij}, \lambda, \omega_i^n +$ 
36:                $1, \beta, L_i^n)$ 
37:              $\mathcal{L} := \mathcal{L} \cup \{L_j^m\}$ ;  $\text{PUSH}(\mathcal{Q}, L_j^m)$ 
38:           end if
39:         end if
40:          $M_j := M_j + 1$ ;  $m := M_j$ 
41:          $L_j^m := \text{CREATELABEL}(i, j, g_i^n, \underline{q}_i^n, \Delta, q, t_{ij}, e_{ij}, 0, \omega_i^n, \beta, L_i^n)$ 
42:          $\mathcal{L} := \mathcal{L} \cup \{L_j^m\}$ ;  $\text{PUSH}(\mathcal{Q}, L_j^m)$ 
43:       end for
44:     end for
45:   end while
46:   return NONE, NONE // Node not found

```

^A Check if the subtree of current node contains the category of POI that is needed for the next TW and the subsequent ones. Otherwise, goes to the next element in the queue.

The label L_i^n with the lowest value of f is selected and then the algorithm finds the next TW k that must be performed. A check is performed to verify if the current label entails the arrival to the destination point: if so, we verify whether the index ω_i^n of the last visited TW is at least equal $|\mathcal{W}^R|$. If this is not the case, the EV has not visited yet all the non-avoidable TWs, and thus the current label must be discarded. Otherwise, the current label and the set of all the generated labels are returned and the search terminates.

At this point, if k is not NONE the algorithm checks whether or not there exists at least one node in \mathcal{T}_i of the current node i in which the POI constraint of TW k is satisfied. Then it checks the same for all the TWs in \mathcal{W}^R that must be performed after k .

The algorithm proceeds by considering the nodes j such that $(i, j) \in \mathcal{A}$. A sequence of operations assures that the trip from i_n to j_m is feasible, where m is the index of the m -th copy of j that will be created if all the checks are passed. First the energy

constraints. If the energy required on arc (i, j) is greater than the maximum amount for the EV, we discard this label, and the algorithm proceeds to the next node in $\text{STAR}(\mathcal{G}, i)$. Otherwise, if $e_{ij} < Q - q_{\min}$, we consider charging a greater amount of energy with respect to e_{ij} by looping on β .

At line 25 of Alg. 1, the charging time and the charged energy are computed, given the current SoC q_i^n and the amount of energy required e_{ij} . The two values are then updated on line 26, taking the current SoC and the arrival time.

We then compute a temporary value g_{temp} of the arrival time in j . In case $g_{\text{temp}} > t_{\text{end}}$, the current label is discarded. We then verify if the selected TW k is in \mathcal{W}^R . If so, then the algorithm must satisfy the constraints associated with k . First, if $g_{\text{temp}} > \gamma_k^U$, then the arrival time at j will be after the ending time of k , which is not feasible. If instead, g_{temp} is included in the range $[\gamma_k^L - \varphi, \gamma_k^U]$, then the EV arrives at j during the TW k . In this situation, the user may decide to stop in j , and respect the TW constraints, or to continue driving to the next node h and see if it is possible to respect k in node h . This scenario models the case in which, for instance, instead of respecting the lunch constraints in node j , the user prefers to drive longer and eat at another place. In the case we stay in j to respect TW k , we check whether node j has the POI that is required for k . If so, a label L_j^m is created, imposing that $\omega_j^m := \omega_i^n + 1$ (from j_m on, TW k is respected) and $\lambda_j^m = \max\{0, \gamma_k^L - g_{\text{temp}}\} + t_k^{\min}$. The max function is used to compute how much time in advance the EV arrives in j , so the minimum stopping time imposed from any arc from j_m is λ_j^m . If instead node j does not have any POI of the given category ν_k , we can step over and create a label that goes from i_n to j_m without imposing a minimum stopping time λ_j^m . In this case $\lambda_j^m := 0$ and $\omega_j^m := \omega_i^n$. In both cases, the newly created label L_{j_m} is added to the set of labels \mathcal{L} and pushed to the queue \mathcal{Q} . Finally, the loop on β continues after updating β . Algs 2 and 3 outline the creation of a new label, and its heuristic. Finally, if it is not possible to reach \mathcal{D} respecting all the imposed constraints, the algorithm returns NONE.

Algorithm 2 CREATELABEL function. It creates the label from node a to node b , given the arrival time g in a , the SoC q , the charging time Δ , the charged energy q , the driving time and energy, t, e , the minimum amount of time needed to charge in node b in the next label, the index of the last performed TW ω , the charger additional time β and the previous label L .

```

1: function CREATELABEL( $a, b, g, \underline{q}, \Delta, q, t, e, \lambda, \omega, \beta, L$ )
2:    $g := g + \Delta + t; q := \underline{q} + q - e$ 
3:    $h := \text{HEURISTIC}(\underline{b}, \underline{q}, g)$ 
4:    $\underline{f} := g + f$ 
5:    $L := (a, g, h, f, L, \beta, q, \underline{q}, \lambda, \Delta, \omega)$ 
6:   return  $\tilde{L}$ 
7: end function

```

We modify the previously described A* search algorithm for the **MDPM**. We refer to this algorithm as **AsDM**. We add

Algorithm 3 HEURISTIC function. It returns the estimated remaining time from node i to \mathcal{D} in graph \mathcal{G} , considering the SoC \underline{q} and arrival time g .

```

1: function HEURISTIC( $i, q, g$ )
2:    $\pi_{\text{tw}}(i) := \text{MINSTOP}(g)$ 
3:   if  $q - q_{\min} \geq \pi_{\text{cons}}(i)$  then
4:      $\pi_{\text{ch}}(i) := 0$ 
5:   else
6:      $s_{\max} := \text{MAXSLOPE}(\text{SUBGRAPH}(\mathcal{G}, i))$ 
7:      $\pi_{\text{ch}}(i) := \lceil \pi_{\text{cons}}(i) - (q - q_{\min}) \rceil / s_{\max}$ 
8:   end if
9:   return  $\pi_{\text{dr}}(i) + \max\{\pi_{\text{tw}}(i), \pi_{\text{ch}}(i)\}$ 
10: end function

```

two parameters in the labels, as follows

$$L_{j_m} := (i, g_j^m, h_j^m, f_j^m, p_j^m, \beta_j^m, q_j^m, \underline{q}_j^m, \lambda_j^m, \Delta_j^m, \omega_j^m, \Sigma_j^m, \tau_j^m).$$

The first parameter is Σ_j^m , which represents the total score gained from \mathcal{O} to the current label. For copy j_m ,

$$\Sigma_j^m := \Sigma_i^n + \sigma_j, \quad \Sigma_{\mathcal{O}}^1 := 0.$$

The second parameter is τ_j^m , it represents the arrival time in copy j_m . All the TWs constraints are now satisfied using this parameter instead of the arrival time g . This means, for instance, that $\tau_{\text{temp}} = \tau_j^m + \Delta - t_{ij}$, and every g_{temp} is replaced with τ_{temp} . The minimum waiting time becomes $\lambda_j^m := \max\{0, \gamma_k^L - \tau_j^m\} + t_k^{\min}$. For the origin node we have $\tau_{\mathcal{O}}^1 := t_{\text{start}}$ and $g_{\mathcal{O}}^1 := 0$. Finally, due to the different definition of labels, the function CREATELABEL is replaced by CREATELABELDISCOUNTED, and is outlined in Alg. 4.

Algorithm 4 CREATELABELDISCOUNTED function. It creates the label from node a to node b , given the discounted cost g to a , the SoC \underline{q} , the charging time, Δ , the charged energy q , the driving time and energy, t, e , the minimum amount of time needed to charge in node b in the next label, the index of the last performed TW ω , the charger additional time β , the previous label L , the arrival time τ , and the gained profit Σ .

```

1: function CREATELABELDISCOUNTED( $a, b, g, \underline{q}, \Delta, q, t, e, \lambda, \omega, \beta, L, \tau, \Sigma$ )
2:    $\tau := \tau + \Delta + t; \underline{q} := \underline{q} + q - e$ 
3:    $\sigma := \text{SCORE}(b)$ 
4:    $g := g + \Delta + t - \mu\sigma$ 
5:    $h := \text{HEURISTICDISCOUNTED}(B, \underline{q}, \tau, \Sigma)$ 
6:    $\underline{f} := g + f$ 
7:    $L := (A, g, h, f, L, \beta, q, \underline{q}, \lambda, \Delta, \omega, \Sigma, \tau)$ 
8:   return  $\tilde{L}$ 
9: end function

```

Furthermore, function HEURISTIC, is replaced with HEURISTICDISCOUNTED, described in Alg. 5. The new function takes as input the additional parameter Σ and returns the following discounted estimated time to reach \mathcal{D} :

$$\pi_{\text{u}}^2(i) := \pi_{\text{u}}^1(i) - \mu\Sigma.$$

Thus, the final value of f computed for node \mathcal{D} can be compared to the one computed with **MDPM**.

Algorithm 5 HEURISTICDISCOUNTED function. It returns the estimated time from node i to \mathcal{D} , considering the SoC q , the arrival time g and the gained score Σ

```

1: function HEURISTICDISCOUNTED( $i, q, g, \Sigma$ )
2:    $\pi_{\text{tw}}(i) := \max\{0, \gamma_k^L - g + t_k^{\text{min}}\}$ 
3:   if  $q - q_{\text{min}} \geq \pi_{\text{cons}}(i)$  then // See A
4:      $\tilde{\pi}_{\text{cons}}(i) := 0$ 
5:   else
6:      $s_{\text{max}} := \text{MAXSLOPE}(\text{SUBGRAPH}(\mathcal{G}, i))$ 
7:      $\pi_{\text{ch}}(i) := \frac{\pi_{\text{cons}}(i) - (q - q_{\text{min}})}{s_{\text{max}}}$ 
8:   end if
9:   return  $\pi_{\text{dr}}(i) + \max\{\pi_{\text{tw}}(i), \pi_{\text{ch}}\}(i) - \mu\Sigma$ 
10: end function

```

^A Available energy is potentially sufficient to reach \mathcal{D} .

IV. COMPUTATIONAL EXPERIMENTS

A. Data description and preprocessing

We obtain CS locations and specifications from a company, whose name we cannot disclose due to privacy reasons. The CSs were extracted from a bounding box that goes from 43.55° to 49.05° latitude and from 8.68° to 13.11° longitude, covering parts of Italy, Germany, Austria, Switzerland and the totality of Liechtenstein and San Marino (see Fig. 3).

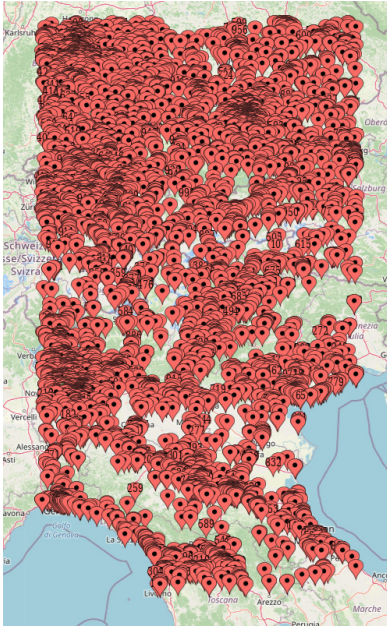


Figure 3: Distribution of the CSs (created with mapcustomizer.com [7]).

CS i has a charging power p_i . We categorize the CSs as follows:

- slow: $p_i \leq 11$ kW,
- medium: $11 \text{ kW} < p_i \leq 30$ kW,
- fast: $p_i > 30$ kW.

We have performed a number of preprocessing actions on the provided data set. The first step was to group CSs at similar locations, for each group the fastest charger within that group

was considered. We then considered that CSs within a radius of $r_M = 100$ m form a cluster and if one CS belongs to two or more clusters, then all of them are merged. This reduction is based on the Haversine formula for computing the distance between each pair of nodes and assumes that the distances are symmetric. Merging all the CSs into a cluster produces a new CS that replaces the other CSs in the cluster. The cluster position is set as the average of the GPS coordinates of the CSs that comprise it, while its charging speed is set as the maximum among the CSs that comprise it. The final dataset is denoted by Γ . The final number of CSs is summarized in Tab. I.

Table I: Distribution of CSs for each country.

	Original # CSs	# CSs after grouping	# CSs after clustering
Germany	59 339	5 955	4 135
Italy	15 009	5 713	4 911
Switzerland	5 400	1 583	1 334
Austria	3 679	1 488	1 219
Liechtenstein	65	33	27
San Marino	34	17	15
Total	83 526	14 789	11 641

Using the post-processed database, we computed the distance and time matrix for each pair of CSs, obtained from multiple requests to the Open Source Routing Machine API (OSRM, [8]) server. The energy required by each arc was instead given by the product of the arc length and the average consumption per kilometer, as in [10].

The sets of required \mathcal{W}^R , and weakly mandatory time windows \mathcal{W}^M are given as input. Furthermore, we consider that tourism AWTs \mathcal{W}^T are also given as input. For $k \in \mathcal{W}^T$ a single CS location \mathcal{P}_k is given. To construct the set \mathcal{S}_k , the Haversine distance is computed by searching for CSs within a radius r_T centered in \mathcal{P}_k . The CSs with a distance less than r_T are included in \mathcal{S}_k . If no eligible nodes are present, the radius is iteratively increased by δ_T and the search is repeated, up to a maximum of \tilde{r}_T . If \mathcal{S}_k is still empty, the computation is resumed and an error informs that it is not possible to reach \mathcal{P}_k unless the EV is left more than \tilde{r}_T away. This process simulates the approach that a user needs to search for a CS near the location she wants to visit. If no CS is available at a reasonable distance from \mathcal{P}_k , then the user can decide to remove or change that tourism stop. For our experiments, we used $r_T = 2$ km, $\delta_T = 200$ m and $\tilde{r}_T = 4$ km.

Given the origin \mathcal{O} , the destination \mathcal{D} and the set of tourism stops \mathcal{W}^T , we construct the set of chargers \mathcal{S} and the set of arcs \mathcal{A} . Using OSRM, the optimal path without charging stops is computed and is used as a lower bound for **SPM**. All the CSs that are within a range of 5 km centered on the OSRM optimal route form the set \mathcal{S} . Finally, we construct \mathcal{A} as the set of all arcs (i, j) connecting nodes in $\mathcal{S}_{\mathcal{O}, \mathcal{D}}$.

In addition, we remove from \mathcal{S} all the nodes that have a distance from \mathcal{O} that is less than r^{min} , due to the fact that we want a solution with a small number of stops. Selecting one of those would have caused the EV to stop for just a few

kilometers from the origin point, which is not desirable. The same reasoning is applied to \mathcal{D} .

We also remove arcs that are going in the opposite direction with respect \mathcal{D} . More precisely, for each arc $(i, j) \in \mathcal{A}$, if by going from i to j the distance to \mathcal{D} is reduced, then (i, j) is kept, otherwise it is deleted. This process nearly halves the amount of edges included in \mathcal{A} .

Since the EV needs to respect the battery capacity Q and we want the total travel time to be small, we consider only the arcs (i, j) such that $r^{\min} \leq d_{ij} \leq r^{\max}$, where

$$r^{\max} := \lfloor (Q - q_{\min})/\eta \rfloor.$$

Finally, nodes that were not reachable from any other CS, as well as the corresponding arcs, are deleted.

B. Experiments and results

The codes were implemented in Python, using IBM ILOG CPLEX Optimization Studio 20.1. The tests ran on a single core of an Apple MacBook Pro with 8 core Apple M1 processor of 3.2 GHz, with 8 GB of LPDDR4 RAM.

The considered EV was a Škoda Enyaq iV 60, with a net battery capacity of $Q := 58 \text{ kWh}$ and a maximum average consumption of $\eta := 0.187 \text{ kWh/km}$ [3]. It has a maximum power charge $P := 40 \text{ kW}$, and we set the minimum required energy level q_{\min} to 15 kWh.

Two subsets of CSs were created: a small one, Γ_1 , with 650 CSs, used to compare the exact solution obtained with the MILP models with the A* search algorithms, and a larger one, Γ_2 , with 5 813 CSs, used to test the A* search. Both datasets are extracted with uniform probability from Γ , so we must guarantee that in each tourism stop there is at least one CS. We created a $5 \text{ km} \times 5 \text{ km}$ square centered in the coordinates of each tourism stop and uniformly extracted four CSs. Γ_1 and Γ_2 were obtained by selecting uniformly a certain amount of CSs from Γ and adding them to the ones selected for the tourism stops.

For both sets of CSs we use the same set of instances defined as trips. Since our CSs lay in a rectangular area that covers part of central Europe, those instances are chosen so that they fully lay in the same geographical area. We generate three main trips, with some variations, such as starting time, presence of tourism stops and minimum stopping times, presence or not of lunch (1 hour) and night (11 hours) ATWs. The ATW of lunch is [12:00, 13:30], while that of the night is [19:00, 22:30]. The instances have the following $\mathcal{O} - \mathcal{D}$ pairs:

- from Genoa to Zürich denoted by GeZu;
- from Livorno to Regensburg denoted by LiRe;
- Stuttgart to Ancona denoted by StAn

The remaining details are summarized in Tab. II. The name of the $\mathcal{O} - \mathcal{D}$ is followed by a running number to distinguish the instances. Overall, we created 20 instances.

We tested the MILP shortest path model, **SPM**, and maximum profit model, **MPM**, as well as both A* heuristics, the A* search algorithm for the **SPM**, **AsM**, and its variant for the maximum discount profit model, **AsDM**, for set Γ_1 . In addition, only the heuristics **AsM** and **AsDM** were applied to

Table II: Description of the instances

ID	Departure	Arrival	Tourism Stops			Lunch Night	
			At	When	Min stop		
GeZu1	10:00	18:30 ⁺¹	Lugano	14:00-17:30	2h	-	-
GeZu2	10:00	18:30 ⁺¹	Lugano	14:00-17:30	2h	Yes	-
LiRe1	10:00	18:30 ⁺¹	-	-	-	Yes	-
LiRe2	10:00	18:30 ⁺¹	-	-	-	-	Yes
LiRe3	10:00	18:30 ⁺¹	Verona	14:00-17:30	2.5h	Yes	-
LiRe4	10:00	18:30 ⁺¹	Verona	14:00-17:30	2.5h	-	Yes
LiRe5	4:00	18:30 ⁺¹	Verona	14:00-17:30	2.5h	Yes	Yes
LiRe6	6:00	18:30 ⁺¹	Verona	14:00-17:30	2.5h	Yes	Yes
LiRe7	8:00	18:30 ⁺¹	Verona	14:00-17:30	2.5h	Yes	Yes
LiRe8	10:00	18:30 ⁺¹	Verona	14:00-17:30	2.5h	Yes	Yes
LiRe9	10:00	18:30 ⁺¹	Verona	14:00-17:30	2h	Yes	Yes
LiRe10	10:00	18:30 ⁺¹	Verona	14:00-17:30	3h	Yes	Yes
StAn1	10:00	18:30 ⁺¹	-	-	-	Yes	-
StAn2	10:00	18:30 ⁺¹	Vaduz	14:00-17:30	2h	Yes	-
StAn3	20:00	18:30 ⁺¹	Bologna	7:00-10:30 ⁺¹	2h	Yes	-
StAn4	10:00	18:30 ⁺¹	Vaduz	14:00-17:30	2h	Yes	Yes
StAn5	10:00	18:30 ⁺¹	Bologna	7:00-10:30 ⁺¹	2h	Yes	Yes
StAn6	6:00	2:00 ⁺¹	Vaduz	9:30-13:00	2h	-	-
			Bologna	18:30-22:00	2h	-	-
StAn7	10:00	22:00 ⁺¹	Vaduz	14:00-17:30	2h	-	Yes
			Bologna	9:30-12:30 ⁺¹	2h	-	Yes
StAn8	13:00	22:00 ⁺¹	Vaduz	14:00-17:30	2h	Yes	Yes
			Bologna	14:00-17:30 ⁺¹	2h	Yes	Yes

set Γ_2 . Then for all **AsDM** models we solved **SPM** imposing that the EV must use the arcs selected by the heuristic. CPLEX was unable to solve any instance in Γ_2 within one hour. We denote by GS_x^y (GT_x^y) the relative gap between the total score (trip time) of the models x and y . Also, to ease the notation, in the following we denote by **AsDM** $_{\mu_0}$ the application of **AsDM** for $\mu = \mu_0$, and use TS for the total score, TT for the trip time, and RT for the run time.

As seen in Tab. II, some instances describe the same trip, with different timings, and thus lead to the same set of unconstrained shortest optimal paths, and to the same set of CSs. For this reason, we subdivide the instances, and for both Γ_1 and Γ_2 extract the set of CSs related to each subdivision and store them. In this way, instances in the same subdivision use the same graph for all the models. In Tab. III we list the size for each subdivision.

Table III: Number of nodes and arcs for each instance.

Instances	Dataset Γ_1				Dataset Γ_2			
	Before		After		Before		After	
	Nodes	ArCs	Nodes	ArCs	Nodes	ArCs	Nodes	ArCs
GeZu1, GeZu2	36	1260	22	52	296	87320	212	3236
LiRe1, LiRe2	43	1806	36	157	480	229920	391	22230
LiRe3 to LiRe10	49	2352	42	244	492	241572	403	23889
StAn1	84	6972	54	276	667	444222	415	19762
StAn2, StAn4	104	10712	73	629	820	671580	555	34622
StAn3, StAn5	83	6806	54	276	666	442890	414	19698
StAn6 to StAn8	103	10506	73	629	819	669942	554	34542

We use the dataset Γ_1 to analyze the performance of the A* search with respect to the exact solution found. Initially we find the shortest path value T^{opt} of **SPM** and **AsM**. Then we compute the maximum relaxed time $T^{\text{max}} = T^{\text{opt}} + T^{\text{add}}$ and solve the **MPM** model. For the evaluation, we fix the value of T^{add} to 1h 30'. The result is then compared to the score gained

with **AsDM** for different values of the score multiplier μ , $\mu = 1, 2$. For the heuristic algorithms, we set $\beta := \{0, 1, 4\}$.

a) *Small dataset*: We solve the **MPM** model and apply the A* algorithm for $\mu = 2$. The results are in Tab. IV and V.

First, the solutions obtained with the A* search are refined using a MILP formulation. In particular, we create another **SPM** in which we set $x_{ij} = 1$ for each arc (i, j) selected in the final solution of **AsDM** for $\mu = 2$. The total trip times obtained by both approaches are quite similar. The results are summarized in Tab. IV and compared with the shortest path model **SPM**. The total trip time increases in average 3.3% when the refined **SPM** is applied, which is a small detour with respect to shortest path.

Table IV: Comparison of total trip time (in hours) between **SPM** and the refined **SPM** in Γ_1

ID	SPM	Ref. SPM	$GT_{\text{SPM}}^{\text{Ref. SPM}}$
	TT	TT	$\times 100$ [%]
GeZu1	8.6	9.0	4.7
GeZu2	9.0	9.0	0.0
LiRe1	14.1	15.7	11.2
LiRe2	23.6	24.4	3.4
LiRe3	15.7	16.1	2.5
LiRe4	24.6	25.5	3.7
LiRe5	30.6	30.9	1.0
LiRe6	28.6	28.6	0.0
LiRe7	26.6	26.7	0.4
LiRe8	25.3	25.4	0.4
LiRe9	25.1	26.2	4.4
LiRe10	27.7	26.6	3.5
StAn1	15.8	17.4	10.1
StAn2	18.2	19.5	7.1
StAn3	16.9	17.3	2.4
StAn4	29.0	29.8	3.8
StAn5	26.3	27.3	3.8
StAn6	19.0	19.3	1.6
StAn7	29.0	29.2	0.7
StAn8	29.8	29.9	0.3
Average			3.3

In Tab. V we compare the total score gained and the run time of **MPM** and **AsDM** for $\mu = 2$. In average, the relative change between the maximum score computed with **MPM** and the one computed with **AsDM** there is an average of 11.9% worse scores with respect to the optimal solution found by **MPM**. This last value is quite large, with some instances having a relative change of over 25%. A possible approach to obtain better results may be to rely on different values of the parameter μ . The run time is quite low for all the models, but it is worth noting that we are considering the small graph created with the CSs in Γ_1 .

b) *Medium dataset*: We want to test the A* search algorithm for the medium dataset Γ_2 . We tested all the instances using only the heuristic approaches, **AsM** and **AsDM**. The results are reported in Tab. VI and VII. According to Tab. VI, we obtain an average increase of the total score with $\mu = 1$ with respect to $\mu = 2$. The difference is due to the fact that with $\mu = 2$ the shortest arcs become more important, even

Table V: Comparison of total score and run time (in sec.) between **MPM** and **AsDM** with $\mu = 2$ in Γ_1

ID	MPM		AsDM₂		$GS_{\text{MPM}}^{\text{AsDM}_2}$ $\times 100$ [%]
	RT	TS	RT	TS	
GeZu1	0.0	7.7	0.0	7.7	0.0
GeZu2	0.0	4.6	0.0	4.6	0.0
LiRe1	0.5	19.2	0.0	12.8	33.3
LiRe2	1.3	16.6	0.0	13.8	16.9
LiRe3	1.2	23.2	0.1	22.6	2.6
LiRe4	1.3	22.6	0.1	18.6	17.7
LiRe5	0.8	22.8	0.0	22.8	0.0
LiRe6	0.6	20.9	0.0	15.1	27.8
LiRe7	0.7	21.6	0.0	21.3	1.4
LiRe8	1.4	22.8	0.0	17.7	22.4
LiRe9	1.5	23.9	0.0	21.4	10.5
LiRe10	0.7	17.4	0.0	14.7	15.5
StAn1	1.4	17.1	0.0	15.2	11.1
StAn2	3.1	18.5	0.0	16.1	13.0
StAn3	1.4	15.2	0.1	11.4	25.0
StAn4	2.6	13.7	0.0	13.3	2.9
StAn5	2.0	17.5	0.2	14.6	16.6
StAn6	0.5	18.2	0.0	18.0	1.1
StAn7	0.3	19.3	0.0	19.2	0.5
StAn8	0.3	19.3	0.0	15.6	19.2
Average	1.1		2.1		11.9

with a lower score in the arrival node. The final solution might improve by tuning μ , also dynamically for each arc. The run time for the **AsM** model is quite high, with an average of 118.8sec for the trip from Livorno to Regensburg and of 81.4sec for the trip from Stuttgart to Ancona. If instead we analyze the **AsDM** model, we see a meaningful drop in the run time, with some exceptions. In particular, the instances that continue to have higher run times are the ones that have large time intervals without any TWs constraints. For instance, after the lunch break **StAn1** has no other TW that constrains the problem, so the research for a best bound solution is more time demanding.

The same holds for **LiRe1**, **LiRe2** and **StAn6**. Instead, **StAn8**, the instance with more TWs, is solved in less than 1 sec. in each discounted model. When the TWs are balanced along the trip, with not too many uncovered time intervals, the computation with the A* search is very fast, even in medium sized graphs. In Tab. VII we can see an average total trip time variation of less than 6.0% in the **AsDM** models with respect to the **AsM**.

V. CONCLUSIONS

We investigate the problem of finding a path for an EV performing a a long trip. The objective is to select CSs that better match the user preferences while respecting ATWs constraints.

We proposed an A* algorithm **AsM** for the shortest path version of our problem. We then proposed the **AsDM** algorithm which privileges POIs preferred by the user. This algorithm was quite fast in finding a shortest path solution with high total score. The algorithm performed fairly well with respect

Table VI: Comparison of total score and run time (in sec.) between **AsM** and **AsDM** with $\mu = 1, 2$ in Γ_2

ID	AsM		AsDM ₁		AsDM ₂		$GS_{AsM}^{AsDM_1}$ ×100 [%]	$GS_{AsM}^{AsDM_2}$ ×100 [%]
	RT	TS	RT	TS	RT	TS		
GeZu1	0.1	8.5	0.0	9.7	0.0	9.7	14.1	14.1
GeZu2	0.1	7.1	0.0	8.5	0.0	9.1	19.7	28.2
LiRe1	0.7	6.2	0.1	22.8	71.9	22.8	267.7	267.7
LiRe2	175.5	11.7	14.0	21.0	25.7	21.0	79.5	79.5
LiRe3	36.5	4.7	0.5	23.2	0.7	23.3	393.6	395.7
LiRe4	77.3	8.6	0.3	27.3	0.4	19.3	217.4	124.4
LiRe5	196.6	11.0	6.8	26.2	4.4	26.2	138.2	138.2
LiRe6	463.7	14.9	22.7	20.0	25.8	20.0	34.2	34.2
LiRe7	213.2	16.5	0.4	27.4	0.3	27.4	66.1	66.1
LiRe8	132.1	8.8	0.3	23.9	0.1	24.5	171.6	178.4
LiRe9	78.6	12.9	0.5	19.4	0.3	23.6	50.4	82.9
LiRe10	128.4	7.6	0.1	23.7	0.4	17.5	211.8	130.3
StAn1	0.7	8.1	134.4	19.2	540.4	19.2	137.0	137.0
StAn2	2.8	5.6	0.5	21.2	0.2	17.0	278.6	203.6
StAn3	99.8	9.7	15.0	19.0	40.2	19.0	95.9	95.9
StAn4	90.6	18.6	62.2	26.1	96.0	26.1	40.3	40.3
StAn5	92.7	12.8	2.7	15.6	4.7	15.6	21.9	21.9
StAn6	30.2	4.1	34.4	14.3	36.5	10.5	248.8	156.1
StAn7	148.9	6.9	37.9	12.9	38.4	19.0	87.0	175.4
StAn8	185.6	11.4	0.2	18.6	0.1	18.8	63.2	64.9
Average	107.7		16.6		44.3		131.9	121.7

Table VII: Comparison of total trip time (in hours) between **AsM** and **AsDM** with $\mu = 1, 2$ in Γ_2

ID	AsM	AsDM ₁	AsDM ₂	$GT_{AsM}^{AsDM_1}$ ×100 [%]	$GT_{AsM}^{AsDM_2}$ ×100 [%]
	TT	TT	TT		
GeZu1	8.6	8.6	8.6	0.0	0.0
GeZu2	8.8	8.9	9.5	1.1	8.0
LiRe1	13.5	14.9	14.9	10.4	10.4
LiRe2	23.1	24.5	24.5	6.1	6.1
LiRe3	15.4	16.8	16.7	9.1	8.4
LiRe4	24.8	26.3	26.3	6.0	6.0
LiRe5	31.0	32.0	32.0	3.2	3.2
LiRe6	28.6	30.0	30.0	4.9	4.9
LiRe7	26.6	27.6	27.6	3.8	3.8
LiRe8	25.0	26.4	26.4	5.6	5.6
LiRe9	24.6	25.5	26.0	3.7	5.7
LiRe10	25.5	26.8	26.9	5.1	5.5
StAn1	15.1	16.6	16.6	9.9	9.9
StAn2	17.5	19.0	19.0	8.6	8.6
StAn3	15.8	17.0	17.0	7.6	7.6
StAn4	27.3	28.7	28.7	5.1	5.1
StAn5	26.2	27.4	27.4	4.6	4.6
StAn6	17.8	19.0	19.2	6.7	7.9
StAn7	28.0	29.5	29.1	5.4	3.9
StAn8	29.7	30.2	31.0	1.7	4.4
Average				5.4	6.0

to the exact solutions. This comparison was only possible on small instances, but the run times tend to increase with the size of the graph. If instead, the chosen trip has many TWs, then the computation is very fast even in medium sized graphs.

The run time of the MILP formulations increases exponentially in the size of the graph, so it can be computationally infeasible to solve even for medium size graphs. Both **AsM**

and **AsDM** solve this problem by storing only the promising states of the EV. However, the heuristics cannot manage real values of the additional charging time, so we must discretize those values using the set β . Solving again the **SPM** model with the arcs selected by the heuristics helps to optimize the charging times on the final solution.

Despite the promising results obtained by **AsM**, **AsDM**, there is potential for further improvements. More efficient heuristics can be developed to account for the potentials, leading to speeding up the A^* search algorithm. Such speeding ups may allow exploring more labels, yielding improved solutions.

REFERENCES

- [1] M. Baum, J. Dibbelt, A. Gamsa, D. Wagner, and T. Zündorf. Shortest feasible paths with charging stops for battery electric vehicles. In *Proceedings of the 23rd SIGSPATIAL international conference on advances in geographic information systems*, pages 1–10, 2015. <https://doi.org/10.1145/2820783.2820826>.
- [2] M. Baum, J. Dibbelt, T. Pajor, J. Sauer, D. Wagner, and T. Zündorf. Energy-optimal routes for battery electric vehicles. *Algorithmica*, 82:1490–1546, 2020. <https://doi.org/10.1007/s00453-019-00655-9>.
- [3] API ChargePrice.com. Open EV Data, Chargeprice.app API. <https://github.com/chargeprice/open-ev-data>. Accessed: 18-03-2022.
- [4] S. Erdoğan and E. Miller-Hooks. A green vehicle routing problem. *Transportation research part E: logistics and transportation review*, 48:100–114, 2012. <https://doi.org/10.1016/j.tre.2011.08.001>.
- [5] E. Fadda, D. Manerba, G. Cabodi, P. Camurati, and R. Tadei. Kpis for optimal location of charging stations for electric vehicles: the biella case-study. In *2019 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pages 123–126. IEEE, 2019. <http://dx.doi.org/10.15439/2019F171>.
- [6] A. Froger, J. E. Mendoza, O. Jabali, and G. Laporte. Improved formulations and algorithmic components for the electric vehicle routing problem with nonlinear charging functions. *Computers & Operations Research*, 104:256–294, 2019. <https://doi.org/10.1016/j.cor.2018.12.013>.
- [7] P. Kaeding. MapCustomizer. <https://www.mapcustomizer.com>. Accessed: 12-03-2022.
- [8] D. Luxen and C. Vetter. Real-time routing with openstreetmap data. In *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, GIS '11*, pages 513–516, New York, NY, USA, 2011. ACM. <https://doi.org/10.1145/2093973.2094062>.
- [9] A. Montoya, C. Guéret, J. E. Mendoza, and J. G. Villegas. A multi-space sampling heuristic for the green vehicle routing problem. *Transportation Research Part C: Emerging Technologies*, 70:113–128, 2016. <https://doi.org/10.1016/j.trc.2015.09.009>.
- [10] A. Montoya, C. Guéret, J. E. Mendoza, and J. G. Villegas. The electric vehicle routing problem with nonlinear charging function. *Transportation Research Part B: Methodological*, 103:87–110, 2017. <https://doi.org/10.1016/j.trb.2017.02.004>.
- [11] M. Schiffer, S. Stütz, and G. Walther. Electric commercial vehicles in mid-haul logistics networks. In *Behaviour of Lithium-Ion Batteries in Electric Vehicles*, pages 153–173. Springer, 2018. https://doi.org/10.1007/978-3-319-69950-9_7.
- [12] M. Schneider, A. Stenger, and D. Goetz. The electric vehicle-routing problem with time windows and recharging stations. *Transportation science*, 48:500–520, 2014. <https://doi.org/10.1287/trsc.2013.0490>.
- [13] T. Vidal, T. G. Crainic, M. Gendreau, and C. Prins. Heuristics for multi-attribute vehicle routing problems: A survey and synthesis. *European Journal of Operational Research*, 231:1–21, 2013. <https://doi.org/10.1016/j.ejor.2013.02.053>.
- [14] T. Zündorf. Electric vehicle routing with realistic recharging models. *Unpublished Master's thesis, Karlsruhe Institute of Technology, Karlsruhe, Germany*, 2014.

Analyzing longitudinal Data in Knowledge Graphs utilizing shrinking pseudo-triangles

Jens Dörpinghaus*, Vera Weil[†], Johanna Binnewitt[‡]

* Federal Institute for Vocational Education and Training (BIBB), Bonn, Germany,
 Email: jens.doerpinghaus@bibb.de, <https://orcid.org/0000-0003-0245-7752>

[†] Department of Mathematics and Computer Science, University of Cologne, Germany,
 Email: weil@cs.uni-koeln.de

[‡] Federal Institute for Vocational Education and Training (BIBB), Bonn, Germany,
 Email: johanna.binnewitt@bibb.de

Abstract—This paper aims to analyze longitudinal data, serial data related to different time points, in knowledge graphs. Knowledge graphs play a central role for linking different data. While multiple layers for data from different sources are considered, there is only very limited research on longitudinal data in knowledge graphs. However, knowledge graphs are widely used in big data integration, especially for connecting data from different domains. Few studies have investigated the questions how multiple layers and time points within graphs impact methods and algorithms developed for single-purpose networks. This manuscript investigates the impact of a modeling of longitudinal data in multiple layers on retrieval algorithms. In particular, (a) we propose a first draft of a generic model for longitudinal data in multi-layer knowledge graphs, (b) we develop an experimental environment to evaluate a generic retrieval algorithm on random graphs inspired by computational social sciences. We present a knowledge graph generated on German job advertisements comprising data from different sources, both structured and unstructured, on data between 2011 and 2021. The data is linked using text mining and natural language processing methods. We further (c) present two different shrinking techniques for structured and unstructured layers in knowledge based on graph structures like triangles and pseudo-triangles. The presented approach (d) shows that on the one hand, the initial research questions, on the other hand the graph structures and topology have a great impact on the structures and efficiency for additional data stored. Although the experimental analysis of random graphs allows us to make some basic observations we will (e) make suggestions for additional research on particular graph structures that have a great impact on the analysis of knowledge graph structures.

I. INTRODUCTION

KNOWLEDGE graphs have been shown to play an important role in recent knowledge mining and discovery, for example in the fields of computational social sciences, digital humanities, life sciences or bioinformatics. They also include single purpose networks (like social networks), but mostly they contain also additional information and data, see for example [1], [2], [3]. Thus, a knowledge graph can be seen as a multi-layer graph comprising different data layers, for example social data, spatial data, etc. In addition, scientists study network patterns and structures, for example paths, communities or other patterns within the data structure, see for example [4]. Very few studies have investigated the questions how multiple layers within graphs impact methods and algorithms

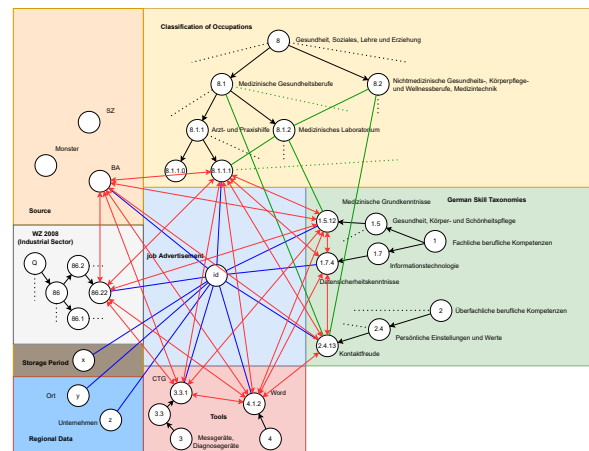


Fig. 1. Knowledge graph representation of German job ads, see Figure 1. In this case the red edges build the pseudo-triangles explaining the context of the job ad.

developed for single-purpose networks, see [5]. In addition, it is possible to store and analyze longitudinal data in knowledge graphs. This is an important topic in medical informatics, for example when working with longitudinal patient records. For example, in [6], the authors use a temporal query language on clinical knowledge graphs. Other authors like [7] use longitudinal patient records within a medical knowledge graph for predictive models. Longitudinal knowledge graphs are related to the versioning of knowledge graphs and other graph-based structures like ontologies. Although research started early on versioning RDF knowledge bases, see [8], only little research has been done on this field. Some attention was paid to the field of the evolution of data structures within information management, see for example [9], [10], [11], and the decentralized collaborative work on knowledge resources, see [12], [13]. Other researchers were interested in parallel world frameworks to analyze scenarios in knowledge graphs, see [14]. Nevertheless, a generic framework for modeling longitudinal data in knowledge graphs is still missing.

In this paper, we focus on an example use case from computational social sciences. In labor market research, ex-

tracking skill requirements from job advertisements (short: job ads) becomes a feasible approach to observe which skills are in demand by employers [15], [16], [17], [18]. Job ads are one way for a company to recruit new employees. Beside general information about the hiring company and the working conditions, they document the current skill needs on the labor market. For a longitudinal view on how skill demands develop, it is necessary to build a model which is capable of not only utilizing these data within a knowledge graph with contextual data, but also for efficient analysis.

We present a detailed overview of the knowledge graph representation in Figure 1. We give a detailed formal definition and overview in the next section.

The main research question of this paper is: How can we model longitudinal knowledge graphs on job ads for an efficient analysis of the development of skills while preserving all contextual data? In order to answer that question, this manuscript investigates both the impact of shrinking triangles in multiple layers and the runtime needed for this approach.

This paper is divided into five sections. The first section gives a brief overview of the research question, state of the art and related work. The second section describes the preliminaries and background. We will in particular introduce knowledge graphs and describe the knowledge graph models on job ads. In the third section, we present the experimental setting and the methods used. The fourth section is dedicated to experimental results and their evaluation. Our conclusions are drawn in the final section.

II. PRELIMINARIES

The term *knowledge graph* (sometimes also called a *semantic network*) is not clearly defined, see [19]. In [20], several definitions are compared, but the only formal definition was related to RDF graphs which does not cover labeled property graphs. However, a *knowledge graph* is a systematic way to connect information and data to knowledge.

Definition 1 (Knowledge Graph). *We define a knowledge graph as mixed graph $G = (E, R)$ with entities $e \in E = \{E_1, \dots, E_n\}$ coming from formal structures E_i , like ontologies.*

By using formal structures within the graph, we are implicitly using the model of a labeled property graph, see [21] and [22]. Here, nodes and edges form a heterogeneous set. Nodes and edges can be identified by using a single label or multiple labels, using a mapping $\lambda : V \cup E \rightarrow \Sigma$, where Σ denotes a set of labels. We need to mention that both concepts are equivalent, since graph databases use the concept of labeled property graphs.

Context is a widely discussed topic in text mining and knowledge extraction since it is an important factor in determining the correct semantic sense of unstructured text. In [23], Nenkova and McKeown discuss the influence of context on text summarization. Ambiguity is an issue for both common language words and those in scientific context. The challenge in this field is not only to extract such context data, but

TABLE I
DIFFERENT LAYERS WITHIN THE KNOWLEDGE GRAPHS OF GERMAN JOB ADS.

	Content	Size	Structure
E_1	Job Ads	ca. 600,000 per year	unstructured
E_2	Classification of Occupations	ca. 18,700	Taxonomy
E_3	Sources	1	unstructured
E_4	German Skill Taxonomy (AMS)	va. 600	Taxonomy
E_5	Tools	ca. 350,000	Taxonomy
E_6	Industrial Sectors	ca. 1,814	Taxonomy
E_7	Storage Period		unstructured
E_8	Regional Data		unstructured

also to be able to store this data for further natural language processing (NLP), like querying and discovery approaches, see for example [4].

In general, for a node $n \in V$, the neighborhood $N(n)$ contains all relevant contextual information. But usually information is best understood using information-triangles. Thus every two nodes $v, w \in N(n)$ form an implicit triangle v, n, w and when adding an additional edge (v, w) this forms a triangle K_3 , see Figure 1 for an illustration. Here, the additional edges that form pseudo-triangles are red.

Definition II.1 (Pseudo-Triangle). *Let $G = (E, R)$ be a knowledge graph and let $n, v, w \in G$ be three nodes in G . Moreover, let $n \in E_i$ for some i and let $v, w \in N(n)$. Then n, v, w form a pseudo-triangle in G .*

The knowledge graph on German job ads is build upon different corpora of job ads from multiple sources. In this paper, we will focus on a corpus from the German Federal Employment Agency. The corpus that we use to extract skills and tools contains approximately 600,000 job ads per year that were advertised from 2011 to 2021. In Table I we present the different knowledge graph layers.

Thus, combing Figure 1 and Table I, it appears that we are working on a knowledge graph $G = (V, E)$ with eight different layers, thus $G = E_1 \cup E_2 \cup \dots \cup E_8 \cup T_1 \cup I_1$ with the given data subsets E_1, \dots, E_8 and the text mining results T_1 and other data integrated in I_1 . We will now discuss how this knowledge graph can be connected with data from different years to build a longitudinal knowledge graph representation.

To test the efficiency of the analysis, we will focus on a very generic question: Given a structured layer which is not constantly changing (e.g. a taxonomy), how do the results on unstructured data (in our case: the job ads) evolve with respect to another structured layer (e.g. another taxonomy, for example tools or skills)? In other words: How can we efficiently retrieve data from a structured layer E_s ordered by another structured layer E_i when both are connected over time by different sets of unstructured data? For the sake of simplicity, we will define $E_s = E_4$ as skills retrieved by text mining and E_i as E_2 given by the classifications of occupations. Both sets are connected by the job ads.

III. METHOD

We can extend the knowledge graph with the information for one particular time point t , in our case for a year: $G^t = E_1^t \cup E_2^t \cup \dots \cup E_8^t \cup T_1^t \cup I_1^t$. With this, we can build a generic graph model comprising multiple times $T = \{t_1, \dots, t_m\}$: $G_T = G^{t_1} \cup G^{t_2} \cup C_{t_1, t_2} \cup G^{t_2} \cup G^{t_3} \cup C_{t_2, t_3} \cup \dots \cup G^{t_{m-1}} \cup G^{t_m} \cup C_{t_{m-1}, t_m}$

In this case, C_{t_i, t_k} comprises all edge relations from G^{t_i} to G^{t_j} . This contains relations like `isEqual` if two entities are equivalent or `isSuccessor` if an entity in G^{t_j} is the successor of an deprecated element in G^{t_i} .

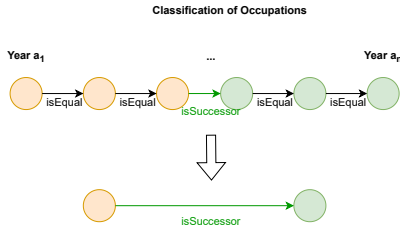


Fig. 2. Merging maximal paths $P = p_1, \dots, p_m$ containing equal data in multiple years.

Thus, a first step to shrink the volume of the knowledge graph is to merge the multiple existence of elements in multiple years. Thus, we search for maximal paths $P = p_1, \dots, p_m$ in G_T where $p_i \in E_j \forall p_i \in P$. The edges between p_i and p_{i+1} are either `isEqual` or `isSuccessor` edges, see Figure 2. Thus for every edge $(p_i, p_{i+1}) \in C_{t_j, t_k}$ we can either merge p_i and p_{i+1} if they are the same (`isEqual`) or leave the `isSuccessor` edges. In our case we are in particular working on E_2 , the classification of occupations.

This can be done with depth-first search, see Algorithm 1, because we explicitly only use the directed subgraph induced by $R = G_T [C_{t_1, t_2} \cup C_{t_2, t_3} \cup \dots \cup C_{t_{m-1}, t_m}]$. The worst-case behavior is in $O(E(R) + V(R))$ and since every node p_i has at most $\Delta(E_2)$ neighbors in E_2 and at most $N(E_1)$ neighbors in E_1 the time complexity of merging the nodes is $O(\Delta(E_2) + N(E_1))$. Thus, the runtime of this step is linear, $O(n)$ in G_T . We denote the graph after step 1 with G_T^1 .

Algorithm 1 STEP-1

Require: Knowledge Graph G_T with layer (tree) E_x^t and mappings $C_{t_i, t_{i+1}}$ for all $t \in T = \{t_1, \dots, t_m\}$

Ensure: Shrunked G_T^1

$$M = N(\bigcup_{i=1, \dots, m} E_x^i) \cup E(\bigcup_{i=1, \dots, m-1} C_{t_i, t_{i+1}})$$

$$2: V = N(\bigcup_{i=1, \dots, m} E_x^i)$$

$$P = \emptyset$$

4: **for** every $v \in E_x^i$ with $v \in V \forall i \in \{1, \dots, m\}$ **do**

$p = \text{DFS}(M, v)$

6: $\text{del}(V, y) \forall y \in p$

$P.\text{add}(p)$

8: **end for**

return $G_T^1 = \bigcup_{i=1, \dots, \text{len}(P)} (p_i \in P)$

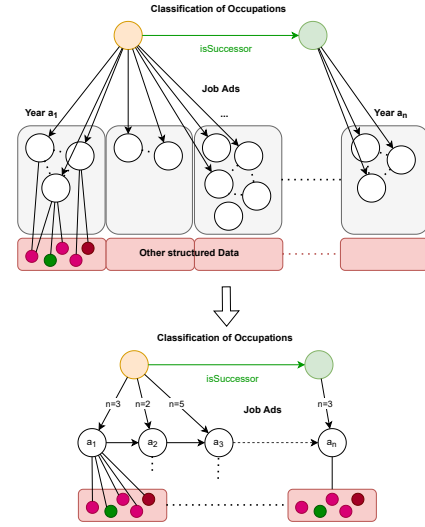


Fig. 3. Merging all job ads for a given year preserving the further links to other structured data.

In general we can make the following observations:

- Since the number of jobs in the classification schema does not increase dramatically, we can assume $N(E_2^t) \approx N(E_2^{t+1})$.
- Thus, even though a number of e_1 jobs may either be deprecated or are added as new items, the size of E_2^1 in G_T^1 is $\max_i N(E_2^i) + e_1 \in O(\max_i N(E_2^i))$.

In a next step, we can merge all job ads for a given year preserving the further links to other structured data. Thus for every time point t and every $v^t \in E_2^t$ we merge all nodes in $N = \{n^t \mid n^t \in N(v^t) \text{ and } n^t \in E_1^t\}$ to a meta node a_t and add an edge (v^t, a_t) with weight $|N|$, see Figure 3. These form pseudo-triangles in E_1 .

The runtime of this step is in $O(tN(E_1)N(E_2))$ and thus is quadratic, $O(n^2)$ in G_T , see Algorithm 2. We denote the changed graph after step 2 with G_T^2 and the new shrunked nodes in E_1 with E_1^2 .

Before continuing with a possible third step of shrinking graph structures, we should consider the theoretical results. Given the question, how the description of skills in job ads evolves in job classifications over the years, in the initial graph G_T we need to consider the following steps:

- Consider the evolution of any classifications v for all times $T = \{t_1, \dots, t_m\}$, runtime $O(m \max_i V(E_2^i))$.
- Consider all skills for any job ad in $N(v^t)$ for all times, runtime $O(V(E_1^t)V(E_4^t))$.

Thus, the average runtime is in $O(n^3)$. For the graph with shrunked pseudo-triangles this reduces to linear runtime:

- Consider any shrunked path p in E_2^1 , runtime $O(\max_i N(E_2^i))$.
- Consider all skills for all times, runtime $O(m)$.

With this third step we can reduce the data complexity, but while in step 1 we do not lose any relevant data, in step 2 we lose the information about specific job ads while preserving

Algorithm 2 STEP-2

Require: Knowledge Graph G_T with an unstructured layer E_x^t and a target layer E_y^t containing paths in P and mappings $C_{t_i, t_{i+1}}$ for all $t \in T = \{t_1, \dots, t_m\}$ and

Ensure: Shrunked G_T^2

$V = \emptyset$

2: $E'_x = \emptyset \forall t \in \{1, \dots, m\}$

$P = \square$

4: **for** every $p \in P$ in E_y^t **do**

for every $p_i \in P = \{p_1, \dots, p_z\}$ **do**

6: **for** every $t \in \{1, \dots, m\}$ **do**

$V' = N(p_i) \cap E_x^t$

8: **add** v' to E'_x

change all edges $(\hat{v}, u) \forall (\hat{v}, u) \in V'$ and $u \notin E_x$,
 $u \notin E_y$ to (v', u)

10: **end for**

12: **end for**

return $G_T^2 = (G_T \setminus E_x) \cup E'_x$

the information for a complete time set. Thus we can make the following observations:

- Structured data like taxonomies and ontologies can be shrunked without any data loss.
- Shrinking unstructured data in triangles or pseudo-triangles always goes along with data loss of particular data points while accumulated information might be preserved.
- Thus, it highly depends on the initial research question which layers and information can be shrunked to improve the runtime of algorithms.

For the given research question, steps 1-2 are the maximum reduction of the initial graph if considering the change for years.

IV. EXPERIMENTAL RESULTS

Our testing environment contains a random graph with m time points containing several graph layers. First, we have a random tree E_2^1 with 18,700 nodes and two probabilities p_p and p_d denoting a rate of a changing predecessor or a deleted node. These probabilities lead to m copies of E_2^1 and their mapping from one time point to the next as described in the previous section.

Second, we generate m times 600,000 random nodes in E_1^1, \dots, E_1^m with equal distributed mappings to E_2^1, \dots, E_2^m as described in the last section. In addition, these nodes receive random edges to 600 descriptive elements. Thus, our experimental setting is highly related to our real-world environment describes in the second section.

We used 50 instances to evaluate the runtime and performance of the algorithms presented in the last section. In Table II and Figure 4 we show the runtime of the two optimization steps. In general, we can see that both steps in average take 0.6 seconds.

TABLE II
RUNTIME OF STEP 1 AND STEP 2 IN SECONDS.

	Step 1	Step 2
Min	0.06	0.09
Max	0.79	0.24
Avg	0.45	0.13

TABLE III
RUNTIME OF RETRIEVAL ALGORITHM ON G_T AND G_T^2 IN SECONDS.

	G_T	G_T^2
Min	0.17	0.01
Max	1.01	0.02
Avg	0.58	0.01

In Table III and Figure 5 we show the runtime of the retrieval described in the last section. We can see that the runtime of both optimization steps is nearly the same as one retrieval on the initial graph G_T . This is not surprising, since the steps are quite similar. The retrieval on the optimized graph G_T^2 is much faster and at latest with the second run of a retrieval algorithm, we see a good improvement of runtime.

V. DISCUSSION AND OUTLOOK

This paper investigates the impact of longitudinal data in knowledge graphs. Knowledge graphs play a central role for linking different data. While multiple layers for data from different sources are considered, there is only very limited research on longitudinal data in knowledge graphs. We presented an experimental environment to evaluate one generic retrieval heuristic given different – both structured and unstructured – data layers. The result clearly shows that the graph structures and topology has a great impact on the efficient retrieval of additional data stored. The initial very generic question was: Given a structured layer which is not constantly changing (e.g. a taxonomy), how do the results on unstructured data (in our case: the job ads) evolve with respect to another structured layer (e.g. another taxonomy, for example tools or skills) evolve? In other words: How can we efficiently retrieve data from a structured layer E_s ordered by another structured layer E_i when both are connected over time by different sets of unstructured data? We specified three example layers to illustrate our optimization approach on (pseudo-)triangles and to evaluate the efficiency.

In particular, we propose a first draft of a generic model for longitudinal data in multi-layer knowledge graphs. This approach stores copies of the knowledge graph on multiple time points and the mapping between nodes in one and a following time point. Since some optimization can be done without losing data, e.g. step 1, we propose further research on a generic longitudinal data model to use these approaches when building the knowledge graph. Second, we develop an experimental environment to evaluate a generic retrieval algorithms on random graphs inspired by computational social sciences. This example was highly influenced by the boundary conditions given by the real-world problem, a knowledge

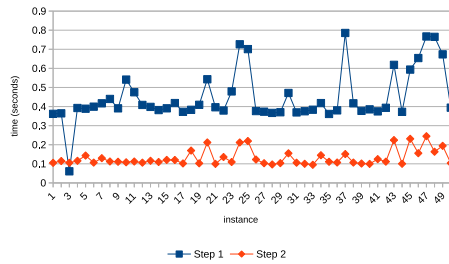


Fig. 4. Resulting runtimes of step 1 and step 2 in seconds.

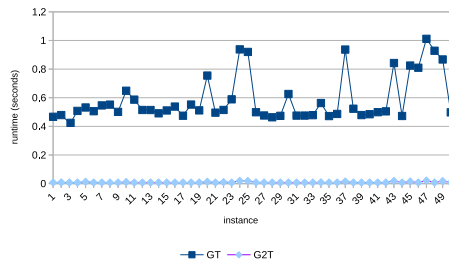


Fig. 5. Resulting runtimes of retrieval algorithm on G_T and G_T^2 in seconds.

graph generated on German job advertisements comprising data from different sources, both structured and unstructured, on data between 2011 and 2021. The data is linked using text mining and natural language processing methods. In general, we present two different shrinking approaches for structured (step 1) and unstructured (step 2) layers in knowledge graphs based on graph structures like triangles and pseudo-triangles. Here, more research needs to follow. While we have argued that these approaches are generic and can be used for any content, further attention for triangles and pseudo-triangles is needed. They form a crucial factor both for understanding the data context and for efficient retrieval of these data.

The presented approach shows that on the one hand the initial research questions (what are the layers to shrink) and on the other hand the graph structures and topology have a great impact on the structures and efficiency for additional data stored. The experimental results show promising results, but further research is necessary to build a generic, time-efficient representation of longitudinal data in knowledge graphs.

REFERENCES

- [1] D. Suárez, J. M. Díaz-Puente, and M. Bettoni, “Risks identification and management related to rural innovation projects through social networks analysis: A case study in Spain,” *Land*, vol. 10, no. 6, p. 613, 2021.
- [2] L. M. Berhan, A. L. Adams, W. L. McKether, and R. Kumar, “Board 14: Social networks analysis of African American engineering students at a PWI and an HBCU—a comparative study,” in *2019 ASEE Annual Conference & Exposition*, 2019.
- [3] C. Rollinger, “Amicitia sanctissime colenda,” *Freundschaft und soziale Netzwerke in der Späten Republik*, 2014.
- [4] J. Dörpinghaus and A. Stefan, “Knowledge extraction and applications utilizing context data in knowledge graphs,” in *2019 Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2019, pp. 265–272.
- [5] G. Rossetti, S. Citraro, and L. Milli, “Conformity: A path-aware homophily measure for node-attributed networks,” *IEEE Intelligent Systems*, vol. 36, no. 1, pp. 25–34, 2021.
- [6] A. Callahan, V. Polony, J. D. Posada, J. M. Banda, S. Gombar, and N. H. Shah, “Ace: the advanced cohort engine for searching longitudinal patient records,” *Journal of the American Medical Informatics Association*, vol. 28, no. 7, pp. 1468–1479, 2021.
- [7] X. Xu, X. Xu, Y. Sun, X. Liu, X. Li, G. Xie, and F. Wang, “Predictive modeling of clinical events with mutual enhancement between longitudinal patient records and medical knowledge graph,” in *2021 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2021, pp. 777–786.
- [8] S. Auer and H. Herre, “A versioning and evolution framework for RDF knowledge bases,” in *International Andrei Ershov Memorial Conference on Perspectives of System Informatics*. Springer, 2006, pp. 55–69.
- [9] F. Zablith, G. Antoniou, M. d’Aquin, G. Flouris, H. Kondylakis, E. Motta, D. Plexousakis, and M. Sabou, “Ontology evolution: a process-centric survey,” *The knowledge engineering review*, vol. 30, no. 1, pp. 45–75, 2015.
- [10] M. Javed, Y. M. Abgaz, and C. Pahl, “Ontology change management and identification of change patterns,” *Journal on Data Semantics*, vol. 2, no. 2, pp. 119–143, 2013.
- [11] Y. Roussakis, I. Chrysakis, K. Stefanidis, G. Flouris, and Y. Stavarakas, “A flexible framework for understanding the dynamics of evolving RDF datasets,” in *International Semantic Web Conference*. Springer, 2015, pp. 495–512.
- [12] N. Arndt, P. Naumann, N. Radtke, M. Martin, and E. Marx, “Decentralized collaborative knowledge management using git,” *Journal of Web Semantics*, vol. 54, pp. 29–47, 2019.
- [13] S. Cardoso, C. Reynaud-Delaître, M. Da Silveira, Y.-C. Lin, A. Gross, E. Rahm, and C. Pruski, “Evolving semantic annotations through multiple versions of controlled medical terminologies,” *Health and Technology*, vol. 8, no. 5, pp. 361–376, 2018.
- [14] A. Eibeck, A. Chadzyski, M. Q. Lim, K. Aditya, L. Ong, A. Devanand, G. Karmakar, S. Mosbach, R. Lau, I. A. Karimi *et al.*, “A parallel world framework for scenario analysis in knowledge graphs,” *Data-Centric Engineering*, vol. 1, 2020.
- [15] M. Stops, A.-C. Bächmann, R. Glassner, M. Janser, B. Matthes, L.-J. Metzger, C. Müller, and J. Seitz, “Machbarkeitsstudie kompetenz-kompass: Teilprojekt 2: Beobachtung von kompetenzanforderungen in Stellenangeboten.” [Online]. Available: <https://www.bmas.de/DE/Service/Publikationen/Forschungsberichte/fb-553-machbarkeitsstudie-kompetenz-kompass.html>
- [16] Bertelsmann Stiftung and Burning Glass Technologies, “Digitalization in the German labor market: Analyzing demand for digital skills in job vacancies.”
- [17] S. Köhne-Finster, I. Leppelmeier, R. Helmrich, D. Deden, A. Geduldig, B. Güntürk-Kuhl, P. Martin, C. Neuber-Pohl, M. Schandock, R. S. Schreiber, and M. Tiemann, *Berufsbildung 4.0 - Fachkräftequalifikationen und Kompetenzen für die digitalisierte Arbeit von morgen: Säule 3: Monitoring- und Projektionssystem zu Qualifizierungsnotwendigkeiten für die Berufsbildung 4.0*, 1st ed., ser. Wissenschaftliche Diskussionspapiere. Leverkusen: Verlag Barbara Budrich, 2020, vol. Heft 214.
- [18] A. Bhola, K. Halder, A. Prasad, and M.-Y. Kan, “Retrieving skills from job descriptions: A language model based extreme multi-label classification framework,” in *Proceedings of the 28th International Conference on Computational Linguistics*, D. Scott, N. Bel, and C. Zong, Eds. Stroudsburg, PA, USA: International Committee on Computational Linguistics, 2020, pp. 5832–5842.
- [19] D. Fensel, U. Şimşek, K. Angele, E. Huaman, E. Kärle, O. Panasiuk, I. Toma, J. Umbrich, and A. Wahler, *Introduction: What Is a Knowledge Graph?* Cham: Springer International Publishing, 2020, pp. 1–10. [Online]. Available: https://doi.org/10.1007/978-3-030-37439-6_1
- [20] L. Ehrlinger and W. Wöb, “Towards a definition of knowledge graphs,” *SEMANTICS (Posters, Demos, SuCCeSS)*, vol. , no. 48, 2016.
- [21] M. A. Rodriguez and P. Neubauer, “The graph traversal pattern,” in *Graph data management: Techniques and applications*. IGI Global, 2012, pp. 29–46.
- [22] —, “Constructions from dots and lines,” *Bulletin of the American Society for Information Science and Technology*, vol. 36, no. 6, pp. 35–41, 2010.
- [23] C. C. Aggarwal and C. Zhai, “An introduction to text mining,” in *Mining text data*. Springer, 2012, pp. 1–10.

Agricultural System Modelling with Ant Colony Optimization

Stefka Fidanova, Ivan Dimov
 Institute of Information and
 Communication Technology
 Bulgarian Academy of Sciences
 Sofia, Bulgaria
 E-mail: stefka@parallel.bas.bg

Denitsa Angelova
 Euro-Mediterranean Center on
 Climate Change
 Ca' Foscari University of Venice
 Venice, Italy
 E-mail: denitsa.angelova@cmcc.it

Maria Ganzha
 System Research Institute
 Polish Academy of Sciences
 Warsaw and Management Academy
 Warsaw, Poland
 E-mail: maria.ganzha@ibspan.waw.pl

Abstract—Cereals contribute significantly to humanity’s livelihood. They are a source of more food energy worldwide than any other group of crops. Their production contributes considerably to the total global anthropogenic greenhouse gas (GHGs) emissions. In this study we propose a basic bio-economic farm model (BEFM) solved with the help of Ant Colony Optimization (ACO) methodology. We aim to assess farm profits and risks considering various types of policy incentives and adverse weather events. The proposed model can be applied to any annual crop.

Index Terms—Agricultural System Modeling, Bio-economic farm model, Ant Colony Optimization, Metaheuristics

I. INTRODUCTION

CEREALS are essential for human nutrition and health. Their production accounts for a substantial amount of the greenhouse gas emission of the agricultural sector, which is in many cases directly affected by a changing climate. The policies of the European Union (EU) recognize these facts. The Farm to Fork Strategy is central to the efforts to decarbonize the economy outlined in the European Green Deal [14]. EU policy encourages agricultural producers to reconcile economic with environmental objectives. Apart from maintaining economic growth, competitiveness, and employment recent policies seek to optimize the use of scarce natural resources such as land and water, to reduce the use of chemical inputs and fertilizers, to promote integrated nutrient management and on- and off-farm biodiversity through sustainable intensification of agricultural systems. The Sixth Report (AR6) of the International Panel for Climate Changes (IPCC) [13], published in August 2021, states that “Each of the last four decades has been successively warmer than any decade that preceded it” (Paragraph A.1.2., page 6 in [13]).

An overview by Reidsma et al. (2018) identifies 202 studies conducted between 2007 and 2015 using bio-economic farm models (BEFMs) [1]. Recent prominent applications include the analysis of cotton production in Uzbekistan [2]. A stochastic dynamic optimization model by Spiegel et al. allows for the representation of managerial flexibility by adopting real options modeling techniques [3]. Another study by Spiegel et al. contributes towards the simultaneous appraisal of investment and managerial behavior and its environmental impacts. The usefulness of the method is demonstrated by an application to

a perennial biomass energy production system [4]. A second application to short rotation coppices can be found in the multi-objective optimization model of Rössert et al. [5]. An overview by Britz et al. determines the desirable features of a BEFM by comparing four generic template-based models selected based on preset criteria [6].

This paper proposes a decision-making system based on a BEFM that has the potential to exhibit many desirable features identified by literature. Cropland allocation is framed as a constrained profit maximization problem from the point of view of the farmer. Apart from the resource constraints the farmer is subjected to various policies and environmental influences. The optimization problem thus includes a lot of constraints and some uncertainties. We apply ACO methodology [7] to solve it. The model is calibrated for Bulgarian crop farms.

II. PROBLEM FORMULATION

The problem is framed as a microeconomics profit maximization problem. Farmers presumably maximize their profit every year to by allocating cropland to a specific annual crop and deciding how intensely to cultivate it. They consider various preconditions, such as various prices, price expectations, expenses, costs, taxes, fees, subsidies and, finally, the probability of an adverse events.

The system relying on profit maximization can be applied in the presence of one or more plots and several suitable for sowing crops. The system considers the expected price for each crop, the average yield for each crop for each of the plots. Included are the subsidies for the individual crops that the farmer can receive, as well as the costs for each of the crops related to tillage, planting, fertilization, crop cultivation, harvesting, etc., as well as the costs for each of the plots posed by rent, taxes and fees. Included is the probability of an adverse event reducing yields such as ozone pollution, cold winter, drought, etc. and a coefficient showing the reduction in yield if the event occurs.

The result of the optimization is a recommendation to the farmer with respect to the cropland allocation and the cultivation intensity so that the profit is maximized.

III. SOFTWARE AND DATA

Software implementing the decision-making system has been developed. The input parameters are: plots of land; crops for planting; minimum area for sowing crops, on each of the plots; expected price of every crop; subsidy per unit area for every crop; yield per unit area for every crop from every plot; estimated cost per unit area for crop (amount of tillage costs, seeds, fertilizers, chemicals, labor, overheads); costs per plot, independent of the crop sown (land lease, taxes, etc.); likelihood of an adverse event reducing the harvest (ozone pollution, drought, hail, pests, etc.); a coefficient showing the reduction in yield as an adverse event appears.

The decision-making system is calibrated using price and yield data provided by the Bulgarian Chamber of Agriculture. The used crops are the cereals most planted in Bulgaria (corn, sunflower, wheat, barley). The yield data refer to North-East Region in Bulgaria for the year 2018. The prices for crops, subsidies and costs are measured in Euro, while the land is measured in decares.

IV. ANT COLONY OPTIMIZATION ALGORITHM

Nature does not tolerate extravagance. It has millions of years of experience. It can teach us how to achieve maximum results with minimal effort. That is why methods with ideas from nature are so successful.

The ACO is a methodology, which is nature-inspired. It belongs to the group of metaheuristics. The method follows the real ants behavior when looking for food. Real ants use pheromone substance, to mark their path and to return back. Normally an ant moves in random fashion and when it comes across a previously laid pheromone it decides whether to follow it and reinforce it with an additional pheromone. So the more ants follow a given trail, the more attractive that trail becomes. Pheromone evaporates over time. Thus the pheromone level of less/not used paths decreases and they become less desirable later. It prevents the ants from following wrong and useless paths. Observations show that ants manage to find the shortest path between the nest and the food source using only the concentration of the pheromone, i.e. their collective intelligence.

A. Main ACO Algorithm

Problems with strict restrictions and a large number of parameters usually require a lot of computing resources. An option is to be applied some metaheuristics. They are more flexible and fast at the expense of accuracy [7].

For a first time, ant behavior is used for solving optimization problems by Marco Dorigo [8]. Later some modifications are proposed, mainly in pheromone updating rules [7]. The basic in ACO methodology is graph representation of the problem and simulation of ants behavior. The solutions are represented by paths in a graph and the aim is to find shorter path corresponding to given constraints. The requirements of ACO algorithm are as follows: Appropriate representation of the problem by a graph; Appropriate pheromone placement on the nodes or on the arcs of the graph; Suitable

problem-dependent heuristic function, which manage the ants to improve solutions; Pheromone updating rules; Transition probability rule, which specifies how to include new nodes in the partial solution; Appropriate algorithm parameters.

The transition probability $P_{i,j}$, is a product of the heuristic information $\eta_{i,j}$ and the pheromone trail level $\tau_{i,j}$ related to the move from node i to the node j , where $i, j = 1, \dots, n$.

$$P_{i,j} = \frac{\tau_{i,j}^a \eta_{i,j}^b}{\sum_{k \in \text{Unused}} \tau_{i,k}^a \eta_{i,k}^b}, \quad (1)$$

where Unused is the set of unused nodes of the graph.

The initial pheromone level is the same for all elements of the graph and is set to a positive constant value τ_0 , $0 < \tau_0 < 1$. After that at the end of the current iteration the ants update the pheromone level [7]. A node become more desirable if it accumulates more pheromone.

The main update rule for the pheromone is:

$$\tau_{i,j} \leftarrow \rho \tau_{i,j} + \Delta \tau_{i,j}, \quad (2)$$

where ρ decreases the value of the pheromone, which mimics evaporation in a nature. $\Delta \tau_{i,j}$ is a new added pheromone, which is proportional to the quality of the solution. For measurement of the quality of the solution is used the value of the objective function of the ants solution.

The first node of the solution is randomly chosen. With the random start the search process is diversifying and the number of ants may be small according the number of the nodes of the graph and according other population based metaheuristic methods. The heuristic information represents the prior knowledge of the problem, which is used to better manage the algorithm performance. The pheromone is a global history of the ants to find optimal solution. It is a tool for concentration of the search around best so far solutions.

B. ACO for Agricultural Modeling

ACO algorithm is a constructive method. Every ant constructs its solution, taking in to account the constraints. In our application an ant chooses first crop randomly between the crops for sowing and assign it to the randomly chosen possible plot. The assigned land is equal to the minimal lend for this crop. The next crop and plot is chosen applying probabilistic rule called transition probability. If on the chosen land this crop is assigned yet we increase the land with 1. If the number of assigned crops for all lends is less then the minimal number, than unassigned crops have two times higher probability to be assigned, or:

$$P_{i,j} = \begin{cases} \frac{\tau_{i,j}^a \eta_{i,j}^b}{\sum_{k \in \text{Unused}} \tau_{i,k}^a \eta_{i,k}^b} & \text{more crops or crop } i \text{ is assigned} \\ 2 \frac{\tau_{i,j}^a \eta_{i,j}^b}{\sum_{k \in \text{Unused}} \tau_{i,k}^a \eta_{i,k}^b} & \text{less crops, crop } i \text{ is'nt assigned} \end{cases} \quad (3)$$

We construct the following heuristic information:

$$\eta_{i,j} = PO \times Decr_{i,j} \times d_{i,j} \times P_i + (1 - PO) \times d_{i,j} \times P_i + (S_i - r1_{i,j}) \times N_{ij}$$

where PO is a probability for adverse event, $Decr_{i,j}$ is decrease of yield if adverse event appear, $d_{i,j}$ is output for crop i from land j , P_i is expected price of crop i , S_i is subsidy for crop i , $r1_{i,j}$ is expenses for crop i from land j , N_{ij} is the sown area of crop i on land j . Thus the adverse event and its influence is taken in to account. When the probability the adverse event to appear is more than 0, then the output of crop i decrease with coefficient $Decr$. For different crops this coefficient is different, because the adverse event influences different crops in different way. For example the corn and sunflower are more influenced by drought, than wheat and barley.

The used objective function is as follows:

$$F = \sum_{i=1}^n \sum_{j=1}^M PO \times Decr_{i,j} \times d_{i,j} \times P_i \times N_{ij} + (1 - PO) \times d_{i,j} \times P_i \times N_{ij} + (S_i - r1_{i,j}) \times N_{ij}$$

Thus the objective function takes in to account probability of adverse event, different expenses, prices of the crops and sown.

V. COMPUTATIONAL RESULTS AND DISCUSSION

Preparing test cases is a complex task. They need be as realistic as possible in order to draw the right conclusions, but also be able to show the qualities and capabilities of the proposed algorithm.

The proposed bio-economic farm model is tested on a test problems with following common parameters:

TABLE I: Test instances characteristics

Parameters	Value
plots of land	{100, 200}
crops	4
minimal area	10
minimal number of crops	4

The parameter settings of our ACO algorithm is shown in Table II and are fixed experimentally after several runs.

TABLE II: ACO parameter settings

Parameters	Value
Number of iterations	100
ρ	0.5
τ_0	0.5
Number of ants	20
a	1
b	1

Several test instances are prepared. The baseline of the bio-economic model refers to the situation without any subsidy or adverse event. Four scenarios are constructed to showcase the capabilities of the bio-economic model: a subsidy for barley amounting to 11 Euro per decare; with subsidy for both barley 11 Euro per decare and wheat 10 Euro per decare; an 80%

probability of drought that decreases corn and sunflower yields with a coefficient of 0.5 for corn and 0.7 for sunflower; a subsidy for barley 30 Euro per decare.

TABLE III: Baseline without any subsidy or adverse events

land	wheat	corn	sunflower	barley
Land1	0	10	90	0
Land2	10	140	40	10

When there is no adverse event and subsidy the corn and sunflowers are preferable because they are more profitable than the two other crops, Table III.

TABLE IV: Subsidy for barley 11 Euro per decare

land	wheat	corn	sunflower	barley
Land1	0	10	90	0
Land2	10	140	40	10

The situation is not changed when the subsidy of the barley is 11 Euro per decare, Table IV. It means that this subsidy is not effective as an incentive to make the barley attractive to the farmer.

TABLE V: Subsidy for barley 11 Euro per decare, for wheat 10 Euro per decare

land	wheat	corn	sunflower	barley
Land1	0	10	90	0
Land2	50	140	0	10

We observe that when the barley and wheat are subsidized the wheat becomes more attractive than the sunflower for the Land2, Table V. It means that a subsidy of 10 Euro per decare is sufficient to make the wheat more attractive.

TABLE VI: 80% probability for drought

land	wheat	corn	sunflower	barley
Land1	10	0	10	80
Land2	140	10	0	50

The drought influences only corn and sunflower yields, thus with a non-zero probability of drought the two other crops become more attractive while the income of the farmer is preserved, Table VI.

The last scenario envisions a larger subsidy for barley. We observe that the subsidy is effective as an incentive to make the barley attractive to the farmer, Table VII.

With these five examples we showcase the capabilities of the basic bio-economic farm model. In the baseline scenario, without any subsidy, the crops with the higher profit are more attractive to farmers. We show that subsidies can incentivize the farmers to alter behavior and stimulate the cultivation of crops that are not profitable in the market sense. We show how the potential for adverse weather events such as droughts can

TABLE VII: Subsidy for barley 30 Euro per decare

land	wheat	corn	sunflower	barley
Land1	0	10	90	0
Land2	10	140	0	50

influence profits by affecting yields and that the anticipating farmers adapt to it by changing the crops they sow as to maintain their profit level.

The proposed model showcases the influence of selected policies and adverse weather events on cropland allocation and can be adapted for any annual crop. The model can be extended to cover diverse other instruments such as subsidies, taxes, production quotas, guaranteed prices as well as regulations regarding the land management. It can also be extended to describe the production technology of the farmer in detail in particular with respect to the input and output requirements so that it is possible to include environmental considerations in the optimization problem, either as constraints or as desirable outcome of the optimization, e.g. achieving a specific soil carbon balance while maintaining agricultural profits or reducing the use of chemical inputs while maintaining a certain production quota.

This technological detail would also allow us to incorporate emission accounting in the model. This would give us the opportunity to quantify the amount of greenhouse gas (GHG) emissions that are consistent with the production level and the maximum profit. The model we propose can also be coupled to a crop growth simulation model for a more detailed representation of the nutrient balance in the soil or to a hydrological model to model the effects of an adoption of irrigation technology. Extensions are possible with respect to the time horizon of optimization even for annual crops to account for situations such as long-term contracts of the farmer with guaranteed prices. The risk-preferences of the farmer could be considered similar to different management options such as crop rotation. Management flexibility and long-term investment decisions could be incorporated.

VI. CONCLUSION

In this paper we propose a decision-making system based on a generic BEFM for annual crops, which has the potential to exhibit many desirable features of a BEFM identified by literature. The model is solved via Ant Colony Optimization methodology. We showcase the possibilities offered by our system by preparing various scenarios. The system can be used to assist farmers with cropland allocation, but also by the state and by the European administration to simulate the effects of technology adoption and for impact evaluation of policies. It will be further developed, thoroughly tested, and compared

to existing BEFM to validate the model results, but also to promote the development of an ecosystem of BEFM modelers.

AUTHOR CONTRIBUTIONS

All authors have equal contribution. All authors have read and agreed to the published version of the manuscript.

ACKNOWLEDGMENT

The presented work is partially supported by the grant No BG05M2OP011-1.001-0003, financed by the Science and Education for Smart Growth Operational Program and co-financed by European Union through the European structural and Investment funds. The work is supported too by National Scientific Fund of Bulgaria under the grant DFNI KP-06-N52/5 and bilateral project IC-PL/01/2022-2021.

REFERENCES

- [1] Reidsma P., Janssen S., Jansen J., van Ittersum M. K., *On the development and use of farm models for policy impact assessment in the European Union – A review*, Agricultural Systems Vol. 159, 2018, 111-125.
- [2] Djanibekov U., Finger R., *Agricultural risks and farm land consolidation process in transition countries: The case of cotton production in Uzbekistan*, Agricultural Systems Vol. 164, 2018, 223-235.
- [3] Spiegel A., Severini S., Britz W., Coletta A., *Step-by-step development of a model simulating returns on farm from investments: the example of hazelnut plantation in Italy: The example of hazelnut plantation in Italy*, Bio-based and Applied Economics Vol. 9, 2020, 53-83.
- [4] Spiegel A., Britz W., Djanibekov U., Finger R., *Stochastic-dynamic modeling of farm-level investments under uncertainty*, Environmental Modeling and Software Vol. 127, 2020, 1-14.
- [5] Rössert S., Gosling E., Gandorfer M., Knoke T., *Woodchips or potato chips? How enhancing soil carbon and reducing chemical inputs influence the allocation of cropland*, Agricultural Systems Vol. 198, 2022, 1-16.
- [6] Britz W., Ciaian P., Gocht A., Kanellopoulos A., Kremmydas D., Müller M., Petsakos A., Reidsma P., *A design for a generic and modular bio-economic farm model*, Agricultural Systems Vol. 191, 2021, 1-14.
- [7] Dorigo M., Stutzle T., *Ant Colony Optimization*, MIT Press, 2004.
- [8] Bonabeau E., Dorigo M. and Theraulaz G., *Swarm Intelligence: From Natural to Artificial Systems*, New York, Oxford University Press, 1999.
- [9] Fidanova S., Roeva O., Paprzycki M., Gepner P., *InterCriteria Analysis of ACO Start Strategies*, Proceedings of the 2016 Federated Conference on Computer Science and Information Systems, 2016, 547-550.
- [10] Fidanova S., Luqu G., Roeva O., Paprzycki M., Gepner P., *Ant Colony Optimization Algorithm for Workforce Planning*, FedCSIS'2017, IEEE Xplorer, IEEE catalog number CFP1585N-ART, 2017, 415-419.
- [11] Roeva O., Fidanova S., Luque G., Paprzycki M., Gepner P., *Hybrid Ant Colony Optimization Algorithm for Workforce Planning*, FedCSIS'2018, IEEE Xplorer, 2018, 233-236.
- [12] Mucherino A., Fidanova S., Ganzha M., *Introducing the environment in ant colony optimization*, Recent Advances in Computational Optimization, Studies in Computational Intelligence, Vol. 655, 2016, 147-158.
- [13] IPCC, 2021: Summary for Policymakers. In: Climate Change 2021: The Physical Science Basis. Contribution of Working Group I to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change, Masson-Delmotte, V., P. Zhai, A. Pirani, S. L. Connors, C. Péan, S. Berger, N. Caud, Y. Chen, L. Goldfarb, M. I. Gomis, M. Huang, K. Leitzell, E. Lonnoy, J.B.R. Matthews, T. K. Maycock, T. Waterfield, O. Yelekçi, R. Yu and B. Zhou (eds.). Cambridge University Press, 2021.
- [14] European Union policies, https://ec.europa.eu/info/strategy/priorities-2019-2024/european-green-deal_en?msclid=86feef9bbafb11ec8c045931e097187b

A GPU approach to distance geometry in 1D: an implementation in C/CUDA

Simon B. Hengeveld* A. Mucherino,*

*IRISA, University of Rennes 1, Rennes, France.

simon.hengeveld@irisa.fr, antonio.mucherino@irisa.fr

Abstract—We present a GPU implementation in C and CUDA of a matrix-by-vector procedure that is particularly tailored to a special class of distance geometry problems in dimension 1, which we name “paradoxical DGP instances”. This matrix-by-vector reformulation was proposed in previous studies on an optical processor specialized for this kind of computations. Our computational experiments show that a consistent speed-up is observed when comparing our GPU implementation against a standard algorithm for distance geometry, called the Branch-and-Prune algorithm. These results confirm that a suitable implementation of the matrix-by-vector procedure in the context of optic computing is very promising. We also remark, however, that the total number of detected solutions grows with the instance size in our implementations, which appears to be an important limitation to the effective implementation of the optical processor.

I. INTRODUCTION

THE Distance Geometry Problem (DGP) asks whether a simple weighted undirected graph $G = (V, E, d)$ can be realized in the Euclidean space \mathbb{R}^K , with $K > 0$, so that the distance constraints

$$\forall \{u, v\} \in E, \quad \|x_u - x_v\| = d_{u,v},$$

are satisfied [5]. When this is the case, we say that the mapping $x : v \in V \rightarrow x_v \in \mathbb{R}^K$ is a *valid* realization of the graph G . Depending on the DGP application at hand, the vertices $v \in V$ can represent different kinds of *objects*, for which possible positions in \mathbb{R}^K are searched. The edge set E encodes the information about the distance between vertex pairs, and the numerical value of these distances is given by the associated weight. Notice that $\|\cdot\|$ is the Euclidean norm. We suppose that the available distance values are *exact*, i.e. extremely precise.

In this work, we focus our attention of DGPs in dimension 1. In 1979, Saxe proved that the DGP is NP-complete when K is set to 1 [10]. The main DGP application in this dimension is the clock synchronization problem: given a set of clocks (represented by the vertices $v \in V$), and a subset of offset measurements between pairs of clocks (encoded by the edges $\{u, v\} \in E$ and the associated weight $d_{u,v}$), the problem asks whether it is possible to know the precise time indicated by all clocks [11]. This problem is fundamental for the synchronization of events in distributed systems [1], [12], as for example in wireless sensor networks [14].

The Branch-and-Prune (BP) algorithm was proposed in [4] for a subclass of DGP instances that admit the discretization

of their search space. In the 1-dimensional case, this algorithm can be employed under the much weaker assumption that the graph G is connected [9]. In this case, in fact, a vertex order on V , which ensures that every vertex v has at least one predecessor u (exception made for the first vertex in the order), can be easily constructed [8]. This vertex order is indicated by the subscripts associated to the vertices in the discussion below.

We are interested in a particular subclass of DGPs in dimension 1: the class of *paradoxical* DGP instances [2]. These instances are represented by graphs G that are cycle graphs, for which a vertex order on its vertex set can be trivially identified. The paradoxical character of these instances is given by the two following observations. On the one hand, the construction of solutions to these instances appears to be relatively *easy*, because most vertices $v_k \in V$ only depend on one predecessor in the vertex order, so that, for each x_{k-1} , the two new positions $x_{k-1} - d_{k-1,k}$ and $x_{k-1} + d_{k-1,k}$ can be easily computed for v_k . Notice that this procedure allows us to build up a binary tree collecting, on each of its layers, the possible positions for each vertex v_k . On the other hand, however, the absence of any other distance information (apart the distance to the predecessor $d_{k-1,k}$) up to the layer n , where $n = |V|$, makes this class of instances actually *very hard*. In fact, it is only at layer n that the distance between the first vertex v_1 and the last vertex v_n can be exploited to select the only two valid realizations out of a set of 2^{n-1} potential solutions [6].

In this work, we consider the matrix-by-vector reformulation of paradoxical DGPs in dimension 1 (recently proposed in [2] for solving paradoxical DGP instances on a new optical processor) and we implement it on a GPU device. The presented computational experiments show a consistent speed-up when our GPU implementation is compared against the BP algorithm, as well as when the comparison is performed against the sequential implementation of the matrix-by-vector procedure itself. These results confirm therefore that the matrix-by-vector reformulation is promising in the context of optic computing. Our experiments also point out, however, a possible limit in the actual implementation of the optical processor.

The rest of this paper is organized as follows. In Section II, we will describe the matrix-by-vector reformulation of our paradoxical DGP instances in dimension 1. In Section III, we will present our GPU implementation for the matrix-by-



Fig. 1. The pattern given by the signs of the elements of the matrix M . In dark blue, the elements that have positive sign; in light gray the ones having negative sign. Notice that M is here transposed to let it better fit the page.

vector multiplication, which will benefit of some simplifications implied by the particular problem we aim to solve. We will present and discuss our computational experiments in Section IV, and finally draw our conclusions in Section V.

II. A MATRIX-BY-VECTOR REFORMULATION

When the BP algorithm mentioned in the Introduction is employed for the solution of paradoxical DGP instances in dimension 1 [9], a binary tree containing all possible vertex positions can be recursively constructed, and the valid realizations can be selected at the very end when positions are computed for the last vertex $v_n \in V$. Our paradoxical instances have the particularity of solely executing the branching phase of the algorithm until a leaf node of the tree is reached; it is only at this point that the pruning mechanism is invoked, where the only distance not used for branching, the distance related to the edge $\{1, n\}$, is verified. If the distance is satisfied by the current position for v_n , then the path from the tree root to the current node is a valid realization; it can be discarded otherwise.

As remarked in [2], it is possible to replace, for our paradoxical instances, the pruning phase occurring only at layer n with an additional branching phase, which is performed over the fictive vertex v_{n+1} that is introduced in the original graph. The fictive vertex is connected to its predecessor v_n by an edge having the same weight as the original edge $\{1, n\}$. The edge from v_1 to v_n is thereafter removed, breaking in this way the original cycle structure. The main reason for making this manipulation on G is that now the distance information is equally distributed over the vertices of the graph, and the solver of paradoxical instances can perform exactly the same operation when stepping from one vertex to its successor. In order to identify the valid realizations, it is finally necessary to verify that $x_1 = x_{n+1}$.

The introduction of the fictive vertex allows us to reformulate the paradoxical DGP in dimension 1 as a matrix-by-vector multiplication [2]. We introduce the matrix

$$M_{ij} = \begin{cases} -1 & \text{if } (i-1)/2^{j-1} \bmod 2 = 0, \\ 1 & \text{otherwise,} \end{cases}$$

and the vector $y_j = d_{j,j+1}$, which contains the distance information related to our paradoxical instance. Thus, the vector $r = My$ contains all possible positions x_{n+1} for all possible solutions. Notice that the index i varies from 1 to 2^n , whereas the index j varies from 1 to n . The feasible solutions to our paradoxical instances are the ones for which $r_i = 0$ (because x_1 is here implicitly set to 0).

We notice that performing the matrix-by-vector multiplication gives an answer to the original decision problem (does it exist a realization such that ...) but it does not directly provide the realizations, i.e. the sequences of positions on the real line for the vertices of G . In order to construct one selected valid realization, as for example the realization encoded by the i^{th} row of the matrix M , the value of each position x_k^i for the vertex $v_k \in V$ (we added a superscript to x to specify the matrix row) can be obtained by performing the following partial sum:

$$x_k^i = \sum_{j=1}^k M_{ij} y_j.$$

The next section describes an ad-hoc GPU implementation of this matrix-by-vector multiplication.

III. A GPU IMPLEMENTATION

Our GPU implementation does not perform *generic* matrix-by-vector multiplications. For this general problem, the reader can refer to some recent (see for example [7], [13]) and very recent (see [3]) publications on the topic. Differently from the cited papers, our implementation takes advantage of the structure of our matrix M to optimize the computations.

First of all, since the elements of our matrix M are either -1 or 1 , we can trivially “move” all distance values from the vector y to the matrix, by paying only attention to the sign to consider for each distance value when placed in a particular row of the matrix. We define therefore this new matrix:

$$M'_{ij} = \begin{cases} -d_{j,j+1} & \text{if } (i-1)/2^{j-1} \bmod 2 = 0, \\ d_{j,j+1} & \text{otherwise,} \end{cases}$$

from which the vector r can be simply computed by summing up all row elements:

$$r_i = \sum_{j=1}^n M'_{ij}.$$

As a consequence, our GPU implementation will only perform sums, and not products of real numbers.

Another important point in our implementation is the procedure to construct the matrix M' , and in particular for the choice of the sign for each matrix element. The rule given in the definition (involving the modulus operator) is simple to understand and to apply, but it can be computationally very expensive to perform for every element of the matrix. For our implementation, we found another, and more efficient, method to identify the sign of every matrix element.

TABLE I

THE COMPUTATIONAL EXPERIMENTS COMPARING THE STANDARD BP ALGORITHM AGAINST THE SEQUENTIAL AND THE PARALLEL IMPLEMENTATIONS OF OUR MATRIX-BY-VECTOR PROCEDURE. COMPUTATIONAL TIMES ARE GIVEN IN SECONDS. ALL USED INSTANCES WERE RANDOMLY GENERATED AND THEY BELONG TO THE CLASS OF PARADOXICAL INSTANCES. SIMILAR RESULTS CAN BE OBTAINED WITH LARGER INSTANCES.

V	BP algorithm		CPU matrix-by-vector		GPU matrix-by-vector	
	#sols	time	#sols	time	#sols	time
20	3	0.012189	3	0.022815	3	0.000437
21	4	0.021719	4	0.048202	4	0.000849
22	8	0.036419	8	0.099945	8	0.001699
23	16	0.067282	16	0.208566	16	0.003494
24	44	0.133229	44	0.435459	44	0.007211
25	82	0.260027	82	0.905198	82	0.014951
26	130	0.498371	130	1.886785	130	0.030999
27	271	0.989594	271	3.905493	271	0.064336
28	515	2.025879	515	8.110146	513	0.133360
29	1074	4.186474	1074	16.831842	1074	0.263456
30	2134	8.036184	2134	34.801628	2046	0.509210
31	3638	15.836381	3638	71.677937	3358	1.006642
32	7613	34.954935	7613	147.561225	6547	2.032326

Fig. 1 shows the sign distribution over the matrices M and M' : all positive elements correspond to the dark pixels, while all negative elements correspond to the light gray pixels. More than one pattern can be identified in these matrices, but one in particular turns out to be very useful for our GPU implementation. If in fact we interpret every gray pixel as a 0 (instead of a -1), whereas the dark pixels still represent 1's, then we can see every matrix row (notice that the matrix is transposed in Fig. 1) as the binary representation of integer numbers spanning from 0 to $2^n - 1$. Moreover, if we consider the big-endian convention for the bit ordering (which is, the less significant bit is on the left side, differently from our standard convention with decimal numbers), then the integer number at row i is simply the predecessor (in integer arithmetic) of the one at row $i + 1$, and it is the successor of the one at row $i - 1$. If the bits of an integer ℓ encode therefore the signs at row i , the bits of the integer $\ell + 1$ simply encode the signs at row $i + 1$.

In our GPU implementation, every thread is in charge of computing the sums for a subset of matrix rows. This subset forms a block of contiguous matrix rows, so that, once each thread has found out its starting value for ℓ , it simply needs to increase it by one unit per time for treating all subsequent rows. Naturally, all row blocks are supposed to have the same size in order to better exploit the power of the GPU device.

After the computation of every row sum, the thread verifies whether this sum is *close enough* to 0. In the case it is true, the thread keeps this information aside (in binary format) and it sends it back to the CPU at the end of the computations. Notice that this information is binary (a valid realization was found or not), because the symmetry properties [6] of our paradoxical instances indicate that the only chance to have two valid realizations treated by the same thread is when all matrix rows are assigned to one unique thread.

IV. COMPUTATIONAL EXPERIMENTS

This section presents some computational experiments where we compare the standard BP algorithm (see Introduction) against our matrix-by-vector procedure, executed both in sequential and in parallel. All programs¹ were written in C, and CUDA was employed for coding the computations on GPU. The experiments were performed on a workstation equipped with an Intel(R) Xeon(R) CPU E5-2609 v3 @1.90GHz, Nvidia GPU GeForce GTX TITAN X graphics card, and running Ubuntu Linux operating system. We compiled our programs with the version 5.4.0 of GCC, and with the version 9.0.176 of CUDA. In all experiments, our execution on GPU was launched with a thread grid comprising 64 blocks, having 512 threads each.

Table I presents some computational experiments where the BP algorithm is compared against the sequential and the parallel implementations of our matrix-by-vector procedure. We considered instances of size ranging from 20 to 32 which were randomly generated so that to satisfy the properties of paradoxical instances. The cardinality $|V|$ of the vertex sets is reported on the first column of the table. We omit to report the cardinality of the edge set E because it always corresponds to $|V|$ in our instances. The BP algorithm was run only on CPU; the matrix-by-vector procedure was run on both CPU (the sequential version) and GPU (the parallel version).

While it is expected (see Introduction and Section III) that every instance only admits two valid realizations, the second column of Table I shows that the number of solutions (#sols) found by the BP algorithm is larger, and it tends to increase with the instance size. Our interpretation for this phenomenon is that, the more the search space increases in size (exponentially with n), the more are the chances to find a realization that is *close enough* to feasibility. The verification of the final distance $d_{1,n}$ is performed with tolerance $\varepsilon = 10^{-4}$ in all experiments, which allows us to take into consideration the possible round-off error propagation. However, the use of this tolerance seems to enlarge *too much* the subset of realizations for which this final distance appears to be satisfied. We remark, however, that a generic tuning on the value of ε that would work for all instances is naturally not possible.

The third column of our table gives the time in seconds necessary for the BP algorithm to fully explore the search space of the given instances. To measure the time, we used the standard C `clock` function. Recall that the search space has size 2^{n-1} .

The fourth and fifth columns of Table I show the results obtained with the implementation of our matrix-by-vector procedure in sequential. While the number of found solutions does not change w.r.t those found by BP, we remark an increase on the computational time. This was expected, because the matrix-by-vector formulation does not exploit the fact that the computations necessary for a given solution can be partially reused for neighboring solutions (i.e. belonging to

¹The interested reader can find an extended version of this article on the online HAL open archive database, where code snippets are also included.

near matrix rows). This is the reason why the BP algorithm works differently. However, the independence of the matrix rows is an essential feature for our GPU implementation.

Finally, the last two columns of the table show the results obtained with our GPU implementation. We point out that we used the `float` primitive type for the distances and the positions (we did the same in the previous two C implementations, in order to obtain results as uniform as possible). We used `unsigned long` types to store the values of the integer ℓ (see Section III). Naturally, this choice limits the instance size n to 64, but has no impact on the experiments we have performed for this work (the decimal representation of 2^{64} is already a quite “large” integer, composed by 20 digits).

The computational time is significantly reduced, when compared to the sequential version of the matrix-by-vector procedure, as well as when the comparison is performed against the original BP algorithm. We notice, however, that the number of found solutions differs with the other implementations for the instances of larger size. This error is due to the way the GPU threads communicate their results to the CPU: this information is in fact binary (solution found / not found) because at most one solution per thread was initially expected to be identified. Apparently, during the computations, more than one solution was instead (wrongly) detected by the same thread, thus leading to a smaller final solution sum.

The reader may wonder why we have decided not to fix this “issue” in our CUDA implementation. Since our work is motivated by the optical processor mentioned in the Introduction, which is supposed to perform the calculations (although in an analog fashion) in a similar way, it seems important to us to point out this drawback of the implementation. This is actually a quite important limitation for the physical implementation of optical processor.

V. CONCLUSIONS

We have presented a GPU implementation of a matrix-by-vector procedure that is particularly tailored for the solution of paradoxical DGP instances in dimension 1. The idea to reformulate this problem as a matrix-by-vector multiplication comes from previous studies on an optical processor, which is specialized for this class of problems.

Our computational experiments show that our GPU implementation is already able to take advantage of the matrix-by-vector reformulation. On our randomly generated paradoxical instances, the pattern shown by our table of experiments is very regular: the GPU implementation is about 16 times faster than the standard BP algorithm. Naturally, much better speed-ups have been achieved on GPUs; in our case, however, the reformulation transforms the problem in a harder one (even if the complexity class remains the same). Nevertheless, the GPU implementation is still able “to do better” than the standard sequential algorithm.

In view of an implementation of the optical processor in [2], we remark that it is likely to suffer of the same effect on the increased number of detected solutions that we have

observed in our computational experiments. This opens new challenging for the conception and development of this kind of alternative computing devices.

Acknowledgments

We are grateful to Caroline Collange for the fruitful discussions and for the authorization to use one of the machines of her research group to perform our computational experiments.

We are also thankful to the three reviewers that provided very helpful comments on this paper. Unfortunately, for the lack of space (this paper was from the beginning meant to be a short contribution), we could not consider all of them in the final version. However, we intend to deposit an extended version of this paper, including some additional details about our CUDA implementation, on the HAL open archives.

This work is partially supported by the international project MULTIBIOSTRUCT funded by the ANR French funding agency (ANR-19-CE45-0019).

REFERENCES

- [1] N.M. Freris, S.R. Graham, P.R. Kumar, *Fundamental Limits on Synchronizing Clocks Over Networks*, IEEE Transactions on Automatic Control **56**(6), 1352–1364, 2010.
- [2] S.B. Hengeveld, N. Rubiano da Silva, D.S. Gonçalves, P.H. Souto Ribeiro, A. Mucherino, *An Optical Processor for Matrix-by-Vector Multiplication: an Application to the Distance Geometry Problem in 1D*, Journal of Optics **24**(1), 015701, 2022.
- [3] K. Isupov, *Multiple-Precision Sparse Matrix-Vector Multiplication on GPUs*, Journal of Computational Science **61**, 101609, 2022.
- [4] L. Liberti, C. Lavor, N. Maculan, *A Branch-and-Prune Algorithm for the Molecular Distance Geometry Problem*, International Transactions in Operational Research **15**, 1–17, 2008.
- [5] L. Liberti, C. Lavor, N. Maculan, A. Mucherino, *Euclidean Distance Geometry and Applications*, SIAM Review **56**(1), 3–69, 2014.
- [6] L. Liberti, B. Masson, J. Lee, C. Lavor, A. Mucherino, *On the Number of Realizations of Certain Henneberg Graphs arising in Protein Conformation*, Discrete Applied Mathematics **165**, 213–232, 2014.
- [7] A. Monakov, A. Lokhmotov, A. Avetisyan, *Automatically Tuning Sparse Matrix-Vector Multiplication for GPU Architectures*. In: “High Performance Embedded Architectures and Compilers”, Y.N. Patt, P. Foglia, E. Duesterwald, P. Faraboschi, X. Martorell (Eds.), Lecture Notes in Computer Science **5952**, Springer, 111–125, 2010.
- [8] A. Mucherino, *Optimal Discretization Orders for Distance Geometry: a Theoretical Standpoint*, Lecture Notes in Computer Science **9374**, Proceedings of the 10th International Conference on Large-Scale Scientific Computations (LSSC15), Sozopol, Bulgaria, 234–242, 2015.
- [9] A. Mucherino, *On the Exact Solution of the Distance Geometry with Interval Distances in Dimension 1*. In: “Recent Advances in Computational Optimization”, S. Fidanova (Ed.), Studies in Computational Intelligence **717**, 123–134, 2018.
- [10] J. Saxe, *Embeddability of Weighted Graphs in k -Space is Strongly NP-hard*, Proceedings of 17th Allerton Conference in Communications, Control and Computing, 480–489, 1979.
- [11] A. Singer, *Angular Synchronization by Eigenvectors and Semidefinite Programming*, Applied and Computational Harmonic Analysis **30**(1), 20–36, 2011.
- [12] P. Verissimo, M. Raynal, *Time in Distributed System Models and Algorithms*. In: “Advances in Distributed Systems, Advanced Distributed Computing: From Algorithms to Systems”, S.K. Shrivastava, S. Krakowiak (Eds.), Springer, 1–32, 1999.
- [13] V. Volkov, J.W. Demmel, *Benchmarking GPUs to Tune Dense Linear Algebra*, IEEE Conference Proceedings, ACM/IEEE conference on Supercomputing (SC08), 11 pages, 2008.
- [14] Y-C. Wu, Q. Chaudhari, E. Serpedin, *Clock Synchronization of Wireless Sensor Networks*, IEEE Signal Processing Magazine **28**(1), 124–138, 2011.

A Multi-objective Cluster-based Biased Random-Key Genetic Algorithm with Online Parameter Control Applied to Protein Structure Prediction

Felipe Marchi

Santa Catarina State University
Applied Computing Graduate Program
Joinville, SC - Brazil
Email: felipe.r.marchi@gmail.com

Rafael Stubs Parpinelli

Santa Catarina State University
Applied Computing Graduate Program
Joinville, SC - Brazil
Email: rafael.parpinelli@udesc.br

Abstract—The protein structure prediction problem is one of the most important bioinformatics problems. Computational methods can be used to approach this problem and *de novo* methods are able to generate protein structures without the need of having known similar structures to the predicted protein. These methods transform the prediction problem into an optimization problem, using optimization models that combine different energy functions and high-level information. These models usually have only a single optimization objective. However, it is known that this single objective optimization approach may harm the optimization search due to the existence of conflicts between the different terms that compose the optimization objective. The proposed model has three objectives: energy function, secondary structure, and contact maps. A multi-objective Biased Random-Key Genetic Algorithm (BRKGA) with online parameter control, named MOBO, is proposed as the optimizer. The final predictor comprises two phases of the MOBO algorithm and selects a final structure using the MUFOLD-CL clustering method. Results obtained demonstrated that the proposed predictor generated highly competitive results with the literature.

Index Terms—Bioinformatics, Multi-objective Optimization, Evolutionary Computation, Clustering, Online Parameter Tuning, Protein Structure Prediction

I. INTRODUCTION

PROTEINS are base molecules present in living organisms [1]. They are responsible for many biological functions, and understanding their mechanisms is important to better understand how organisms work. One application of this knowledge about proteins is in the biomedicine and pharmaceuticals areas [2], where novel proteins could be developed to combat particular types of diseases.

The Protein Structure Prediction (PSP) problem has the objective of determining the spatial structure of proteins, using the amino acid sequence of a protein as its main input. Although these structures can be determined using classical laboratory methods, such as Nuclear Magnetic Resonance (NMR) or X-ray crystallography [3], they have a considerable cost and are time-consuming. An alternative method is the use of computational resources to simulate these structures.

Recently, the AlphaFold, the deep learning algorithm developed by DeepMind, achieved highly promising results in the CASP13 and CASP14 competition [4]. The AlphaFold combines features derived from homologous templates and from multiple sequence alignment to generate the predicted structure. Nevertheless, AlphaFold has some drawbacks, such as the bias to the PDB database, and the heavy dependency on high-computational efforts for training their model [5]. Also, given the number of possible natural and artificial protein structures, it is currently unfeasible to rely on template-based methods to predict any unknown structure with consistent quality. As such, the PSP is still considered to be an open problem. So far, there is no viable general solution to this challenging problem. In this way, metaheuristics can be a faster option to the problem even though achieved results are not the same as AlphaFold.

In the literature, there are many computational methods proposed as possible approaches to the PSP problem [6], [7]. Among these methods, some works further explore the use of specific types of protein structure information [8], [9].

For a more general solution to this problem, one possible way is to use methods that do not rely on databases. One class of methods that do not employ known structure information are the *ab initio* methods. These methods work by generating structures using some evaluation function, which guides the optimization towards the optimal structure. Different from other types of methods, these methods only need as input the amino acid sequence to work [3].

While it is possible to work with pure *ab initio* methods, which utilize only information provided by the amino acid sequence, their results may be unfeasible given inaccuracies of currently employed energy functions. One way to deal with this issue is to utilize high-level information about the protein structure, such as secondary structure and contact map information. *Ab initio* methods that employ this type of information are called *de novo* methods. This work employs *de novo* methods instead of pure *ab initio*.

All this information related to the PSP problem can be described in an objective function. However, combining multiple information in a single objective function is not always optimal. It is known that some of the terms that compose an energy function, such as the bonded and non-bonded energies, are in conflict [10]. These conflicts indicate that optimizing one particular function term may not optimize the others. Hence, it is interesting to separate conflicting terms in different objectives. The separation of conflicting terms creates a multi-objective model, whose final solution is a set of non-dominating solutions [11]. This set, also called the *Pareto set*, represents possible solutions for the mathematical model.

As the optimization result of a multi-objective problem is a set of non-dominating solutions [11], it may be necessary to employ a decision-making method to select a single final solution from this set. As each non-dominated solution represents some trade-off among all objectives, it is not always trivial to choose a singular solution. To assist this selection, decision-making methods can be employed to filter the non-dominated set and identify the most promising solutions based on some desired properties.

This work is an extension of a previous study [12]. The previous work proposed the use of the Biased Random-Key Genetic Algorithm (BRKGA) method as an optimizer for a *de novo* multi-objective optimization model of the PSP problem. The present work brings some contributions compared to the previous publication:

- Online parameter control strategies were incorporated, creating the new version called MOBO. The main objective of this change is to reduce the number of parameters that the user must define to use the optimizer effectively. This change makes it possible to dedicate more time to problem modeling than parameter tuning.
- The MUFOLD-CL was employed as decision-maker in the Pareto set. The generated clusters were analyzed to determine the distance between the best solution found by the optimizer and the solution selected by the decision-maker.

II. BACKGROUND

A. Proteins and Protein Structure Prediction

Proteins are biomolecules composed of amino acids linked by peptide bonds. The amino acids are molecules composed of a central carbon, called α -carbon, that is connected to an amino group (NH), a carboxyl group (CO), a hydrogen molecule (H), and a side chain [1]. The side chain is a variable group unique to each type of amino acid. The molecular structure of each amino acid is the same, changing only the side chain, which gives its identity.

The structure of a generic amino acid can be seen in Figure 1. These amino acids connect between themselves through the peptide bonds, created by the combination of the carboxyl and amino group [1]. A point of interest in this figure is the three dihedral angles (Φ , Ψ , and Ω), which are important for computational representation of the protein structures.

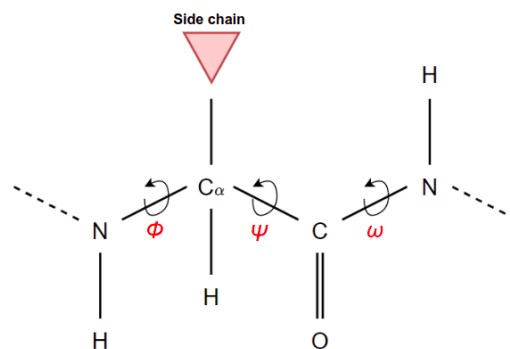


Fig. 1: Chemical structure of an amino acid

1) *Computational representation*: There are several ways to represent a protein structure computationally. The most direct would be to maintain complete information on all the atoms and interactions of the structure. However, this type of representation can be computationally expensive given the large number of atoms and interactions of a single protein structure [3].

One interesting computational representation of protein structures uses only the backbone and side chain torsion angles of each amino acid. Instead of maintaining the entire atomic structure of each amino acid, algorithms may use only the ϕ , ψ , and ω backbone angles and the χ side chain angles. These angles can be seen in Figure 1. By using these angles, the final three-dimensional structure can be uniquely determined.

2) *Structure prediction*: Computational methods can be used to generate protein structures. One class of such methods are the *ab initio* methods, which do not use information from known protein structures [3]. These methods use physical energy functions to guide the generation of protein structures, using only the amino acid as necessary input. As such, *ab initio* methods effectively transform the PSP problem into an optimization problem.

Besides the computational representation of a protein structure, defining a function to evaluate these structures is also necessary. As *ab initio* methods generally perform searches in the space of structures, it is necessary to select a suitable evaluation function. Regarding accuracy, the most optimal choice would be an evaluation function representing the natural energy function. However, such realistic energy functions are computationally expensive [2].

Other types of information can be incorporated into the evaluation function to enhance the search process. Although pure *ab initio* methods do not use information from known structures, it may be beneficial to use this type of information, if available. There are different types of high-level information that can be added to the evaluation function, such as *secondary structure prediction* and *contact maps*. When an *ab initio* method incorporates information other than the energy function, it is called a *de novo* method [3].

B. Biased Random-Key Genetic Algorithm

The Biased Random-Key Genetic Algorithm (BRKGA) [13] is an optimization method of the evolutionary algorithms class. It is a variation of the genetic algorithm in which only the selection and crossover routines are used to evolve solutions [13]. Initially, all individuals are initialized uniformly at random. Each generation, pairs of individuals are selected and combined through a biased uniform crossover operator, where one parent is an elite solution, and the other is a non-elite. A new population is formed by a fraction of the most-fit solutions of the current generation, a fraction of randomly and uniformly generated solutions, and the remaining individuals are the offspring generated through the crossover operation. There is no explicit mutation operator in this algorithm, but an implicit mutation can be simulated through the crossover of a random individual with a non-random individual.

The main aspect of the BRKGA is the clear separation of the problem-dependent and independent parts. Other notable aspects are elitism as a core mechanism for the evolution and the generation of random solutions instead of applying mutation to offspring solutions.

The problem-independent codification is composed of real numbers in the $[0, 1]$ interval, decoded into a problem-specific solution using a decoding function, and evaluated by a fitness function. This way, the algorithm's main structure is standardized for all problems, with only the decoding and fitness functions needing to be developed. This standardization simplifies the algorithm structure by removing the necessity of developing complex solution encodings and various genetic operators, such as crossover or mutation operators.

Similar to other evolutionary algorithms, the BRKGA has a set of parameters that need to be defined. These parameters are the number of iterations (N_{it}), population size (p), elite fraction (p_e), random individuals fraction (p_m), and crossover probability (c_{pr}). Due to the standardized structure of the algorithm, the crossover operator is fixed.

III. RELATED WORK

One of the initial works in the area of multi-objective PSP was [10]. In this pioneering study, the authors proposed a multi-objective model for the PSP problem, decomposing the CHARMM energy function into two objectives: bonded and non-bonded energies. They demonstrated that these energies were in conflict, and combining them in a single objective would harm the optimization. To optimize the model, they used the IPAES algorithm and also employed a decision-making step to select a final structure by using the method of *knees*.

Other works followed this approach of breaking the energy function into bonded and non-bonded energies [14], [15], [16], [17], [18]. Most works approaching the multi-objective PSP employed some evolutionary algorithm [14], [15], [16], [19], [20], [18], [21]. Regarding the protein structure representation, most works employed the full atomic structure [14], [15], [16], [22], [21], although some preferred the centroid representation [19], [20], [18].

Considering the extra information that can be added to complement the energy function, there are several possibilities. Some works employ structural information that energy functions may not capture accurately, such as solvent terms [15], [16], [22], compactness [19], and hydrogen bonding [19], [20]. Also, high-level information that can be predicted was explored, such as protein fragments [19], [20], secondary structures [14], [15], [16], [17], [19], [22], [18], [21], and contact maps [19], [20], [22].

As the complete solution of multi-objective problems is a set of non-dominated solutions, selecting a single final solution from this set may be necessary. Several works employ some decision-making criteria to select the final optimized solution. Popular methods include the *knees* method [10], [23], [14] and clustering-based methods [15], [17], [17], [16], [21]. One promising clustering method for protein structures is the MUFOLD-CL method, which was employed in recent works for the multi-objective PSP problem [17], [21].

IV. PROPOSED MODEL

A multi-objective Biased Random-Key Genetic Algorithm (BRKGA) with online parameter control is proposed, named MOBO. In the present work, proteins are modeled with the Rosetta framework¹ using the full-atom with centroid backbone representation [24], where the side chain is simplified as a centroid. The ϕ , ψ , and ω angles model each amino acid. These angles are in the $[-180, 180]$ domain, except for the ω angle, whose optimal value can be either 180° or 0° [10]. In this work, the ω angle is restricted to 180° , as both values are equally optimal. Figure 2 shows how an amino acid sequence can be coded as a list of angles.

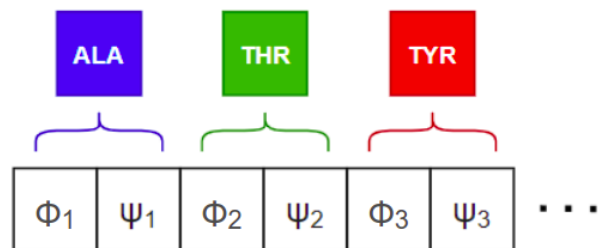


Fig. 2: Protein structure representation as a list of torsion angles.

The optimization model for the PSP problem is composed of 3 objectives, shown in Expressions (1)-(3):

$$\min \text{score4}(\vec{x}) \quad (1)$$

$$\max SS(\vec{x}) \quad (2)$$

$$\max CM(\vec{x}) \quad (3)$$

The \vec{x} vector represents a protein structure coded as a list of angles with size $2L$, where L is the length of the amino

¹<https://www.rosettacommons.org/software>

acid sequence of the protein. The angle list is mapped into a protein structure before being evaluated by each objective function. Expression (1) is the *score4* energy function from the Rosetta framework [24] and is used to evaluate the protein structure.

Expression (2) refers to secondary structure information predicted by the PSIPRED [25] server ². This prediction was performed excluding homologous proteins. Equation (4) details the evaluation, where for each amino acid x_i , the *pSS* function will return the predicted probability for the current secondary structure manifested, determined with the DSSP program [26], which assigns secondary structure to proteins structures. Hence, protein structures with the most probable secondary structures are benefited.

$$SS(\vec{x}) = \sum_i^L pSS(x_i) \quad (4)$$

Expression (3) refers to contact map information predicted by the RaptorX [27] server ³. This prediction was performed by removing homologous proteins. Two atoms are considered in contact if their distance is less than the cutoff distance. Equation (5) describes how each contact is evaluated for a given protein structure.

For each contact c_i in the L -best list, the pair of amino acids c_i^1 and c_i^2 from this contact is verified. If their distance, $d(c_i^1, c_i^2)$, is less than the cutoff value (8\AA), the pair is said to be in contact, and a term equal to the predicted probability of this contact is summed. Otherwise, this probability term is decreased exponentially due to the cutoff value's distance, allowing slight deviations from the cutoff to be considered. Protein structures that exhibit the most probable contacts are rewarded using this objective function.

$$CM(\vec{x}) = \sum_i^L \begin{cases} p_i & \text{if } d(c_i^1, c_i^2) \leq 8 \\ \frac{p_i}{e^{d(c_i^1, c_i^2) - 8}} & \text{otherwise} \end{cases} \quad (5)$$

As global optimizer, the multi-objective BRKGA is employed [12]. The proposed algorithm uses the base structure of the original BRKGA [13] and can be applied to any multi-objective problem that can be approached with evolutionary algorithms. The main modifications were to allow the algorithm to optimize multi-objective problems. Modifications were made to the problem-independent parts to achieve multi-objective optimization. The elitist selection operator from NSGA-II [28] was used to generate the elite part of the population.

An archive (set of solutions) is maintained and updated at each generation. This archive has the same size as the population, and it is updated using the non-dominated sorting with crowding. At the end of the algorithm, the archive is returned and contains the best Pareto set found. However, it may also contain dominated solutions as the best Pareto set may not occupy the entire archive.

²<http://bioinf.cs.ucl.ac.uk/psipred/>

³<http://raptorx.uchicago.edu/>

A. Parameter control

There are several types of parameter control in evolutionary algorithms [29]. This work employs an adaptive online parameter control by simple rules to control and guide the optimizer. The information utilized to guide the search is the population diversity defined in Equation (6):

$$Div(P) = \frac{\sum_{x,y \in P} d_{norm}(x,y)}{|P| * (|P| - 1)} \quad (6)$$

where x and y are individuals from population P , and d_{norm} is the normalized Euclidean distance metric, defined by:

$$d_{norm}(\vec{x}, \vec{y}) = \frac{d(\vec{x}, \vec{y})}{N} \quad (7)$$

where $d(\vec{x}, \vec{y})$ is the Euclidean distance and N is a normalization factor representing the solution space diagonal. This distance is a genotypic diversity metric, that is, it measures the diversity in the problem-independent part of the algorithm. This work will not incorporate fitness information in the parameter control, as the definition of fitness is different for a multi-objective optimizer, and concepts such as best and mean fitness are not trivially defined.

The algorithm is modified to control and increase diversity in the population to incorporate diversity information. In the original BRKGA method, the crossover probability p_{cr} is defined as a parameter of the algorithm and should have a value in the range $(0.5, 1)$. In this work, the crossover parameter is modified to be a uniformly random value in the range $(0.5, 1)$, removing the complexity of defining an optimal value for this parameter and minimizing the impact in the diversity. This value is generated for each crossover operation.

Another modification is the introduction of two diversity control values: the diversity fraction (p_d) and the exploration diversity threshold (δ). These values are used to control the generation of the elite part of the population. The diversity fraction indicates the fraction of the elite part that should be diversified. The exploration diversity threshold is used to indicate the minimum distance between two diversified solutions during the exploration part of the search.

The set P_d of diversified elite solutions is defined as:

$$P_d = \{ \forall \vec{x} \in P_d \mid \Delta(\vec{x}) \geq \delta \} \quad (8)$$

where $\Delta(\vec{x})$ is defined as:

$$\Delta(\vec{x}) = \min_{\substack{\vec{y} \in P_d \\ \vec{x} \neq \vec{y}}} d_{norm}(\vec{x}, \vec{y}) \quad (9)$$

where d_{norm} is the normalized Euclidean distance expressed by Equation (7). This definition of diversified individuals spreads the solutions through the space. Incorporating this diversification in the elite part allows the algorithm to explore the best solutions in different parts of the search space.

Given the population of size p , $E = p \times p_e$ is the size of the elite part of the population and $E_d = E \times p_d$ is the size of the diversified elite. With P_s as the sorted population, the elite part P_e is generated by the following steps:

- While $|P_e| < E$, do:
 - 1) If $|P_d| < E_d$, then:
 - a) If exists $\vec{x}_i \in P_s$ that can be inserted in P_d , then select \vec{x}_i with lowest index i .
 - b) Else, from P_s select \vec{x}_i with greatest Δ
 - c) Insert selected \vec{x}_i in P_d and P_e
 - d) Remove selected \vec{x}_i from P_s
 - 2) Else:
 - a) From P_s select \vec{x}_i with lowest index i
 - b) Insert selected \vec{x}_i in P_e
 - c) Remove selected \vec{x}_i from P_s

The parameter control is performed by dividing the optimization procedure into two phases: exploration and exploitation. The algorithm generates diversified solutions in the exploration phase to search the entire solution space. The algorithm increases the evolutionary pressure in the exploitation phase by favoring the best solutions found. By properly controlling the algorithm parameters, it is possible to balance global and local searches.

To balance between exploration and exploitation, each phase runs for $N_{it}/2$ iterations. For both phases, the parameters p_e and p_m are initially set to 0.25. With these values, half of the population is composed of offspring. With $p_e = 0.25$, it also means that each elite individual should on average generate two offspring each generation.

This initial value was defined empirically. The following initial values and ranges for each parameter were also defined empirically. Although these values may not always be optimal, they are reasonable and straightforward enough to be used as base values.

The exploration phase initializes the optimization process. The diversification parameters p_d and δ are used to explore the search space. The diversification fractions starts as $p_d = 0$, and is modified in steps of $k = 0.01$. This parameter is updated during the exploration phase to keep the population diversity $Div(P)$ above δ . If the current $Div(P) < \delta$, p_d is increased by k . Otherwise, p_d is decreased by k . The parameter p_d is kept in the range $[0, 0.5]$, diversifying at most half of the elite part.

If the $Div(P)$ is still below δ when $p_d = 0.5$, then the parameter p_m is also modified. The parameter is kept in the range $[0.25, 0.5]$ and is updated using step k . By increasing the random part of the population, the diversity of the population is increased. At the maximum value of $p_m = 0.5$, half of the population is uniformly randomly generated each iteration. Also, only 1/4 of the population is offspring, with each elite individual generating on average one offspring and each non-elite parent being on average 2/3 of the time a random individual.

The exploration diversity threshold δ is used to define the diversity of the exploration phase. The user can define it as an algorithm parameter. However, the value 0.4 should be reasonable, in general, and is used as the default value for this parameter. This value was selected due to the nature of uniformly randomly generated solutions and global search.

Considering this, δ is set to 0.4 by default. With this value, the population of the exploration phase has the property of being similar to a uniform distribution without losing the generation of reasonable solutions. Although, in general, it should not be necessary to change this value, in some cases, it can be interesting to lower the parameter to limit the exploration.

In the exploitation phase, the algorithm eliminates the diversity parameters. The parameter p_m is reset to the initial value 0.25 if modified during the exploration. The only parameter modified during this phase is the elite fraction p_e .

The algorithm forces the search to converge to the best solution found during this phase. To increase the convergence, the parameter p_e is increased in the range $[0.25, 0.5]$, using step k . At the maximum value of $p_e = 0.5$, half of the population is composed of elite solutions, and each elite individual will generate, on average, one offspring.

Considering the modifications proposed, the parameters p_e , p_m and c_{pr} are no longer user-defined. The only parameters that still need to be defined are the number of iterations N_{it} and the population size p . The exploration diversity threshold δ can also be defined if necessary.

This proposed parameter control aims to find a reasonable balance between usability complexity and optimization efficiency. The proposed control does not guarantee the selection of optimal values for every problem. However, it should select reasonable values for any general optimization, as it uses the generic concept of exploration and exploitation. Although the algorithm may not execute the most optimal search for some specific problem, the decreased number of parameters allows the user to focus more on the problem modeling and less on the calibration of the algorithm.

B. Parallelism

To increase the scalability of the algorithm, a master-slave model was developed using CPU threads. During fitness evaluation, the master thread divides the population into equal-sized chunks and distributes them to other threads. More formally, considering a number t of threads (e.g., the number of processors), the master will divide the population into chunks of size p/t . Each thread will evaluate the fitness of individuals in its chunk, effectively reducing the processing time of fitness evaluation, which is usually the most time-consuming step of the optimizer.

C. Codification and fitness

The fitness for the multi-objective model of the PSP is a tuple (F_1, F_2, F_3) , where F_1 , F_2 , and F_3 are the objectives defined by the Expressions (1)-(3), respectively. The decoder function is a function that will receive a coded solution $x = (x_1, x_2, x_3, \dots, x_n)$, where n is the size of an encoded chromosome, which is problem-specific, and each x_i is in the domain $[0, 1]$. The decoder should map this chromosome into a problem-specific solution, which will then be evaluated by the fitness function.

In this work, two decoders are used in two different algorithm executions. The first decoder, named fragment decoder, takes an $F = L/9$ -sized chromosome, with L being the amino acid sequence length, and maps it into an angle list by inserting fragments of size nine. Each of these fragments is a continuous sequence of nine amino acids extracted from some known protein structure.

For each amino acid in the protein to be predicted, 200 fragments of size nine were generated using the Robetta server⁴. These fragments were predicted excluding homologous proteins. As the fragments are size nine, each selected fragment contributes nine pairs of ϕ and ψ angles in the solution. The torsion angles ϕ and ψ are extracted from each fragment and used to build the solution.

In the fragment decoder, each variable of the chromosome x (candidate solution) is used to select a fragment from the list of 200 fragments of an amino acid. The x_i variable represents a fragment f_i that starts on the amino acid with position $p_i = 9 \times (i - 1) + 1$ in the protein sequence. The inserted fragment f_i is the one with position $\lceil 200 \times x_i \rceil$ in the fragment list of amino acid p_i . Figure 3a shows the decoding process.

The second decoder, named residue decoder, takes an L -sized chromosome y and maps it into an angle list by extracting the ϕ and ψ angles from each variable. The y_i variable is mapped into torsion angles ϕ_i and ψ_i by using 14 digits as scale, where the first 7 digits are used to generate ϕ_i and the remaining 7 are used to generate ψ_i . If $D_i = \{d_1, d_2, \dots, d_7\}$ are the 7 digits used to generate ϕ_i , the angle ϕ_i can be defined as $\phi_i = -180 + 360 \times r$ where $r = 0.d_1d_2\dots d_7$ (the ψ angle is defined similarly). Figure 3b exemplifies this decoding process.

D. Predictor

The predictor first applies the MOBO with the fragment decoder, MOBO/FRAG, to predict protein structures. This algorithm will search the structure space using fragments, generating valid low-resolution structures. Then, the MOBO with the residue decoder, MOBO/RES, optimizes the initial archive of solutions.

Using the archive from MOBO/FRAG as the initial population, the MOBO/RES can refine the results and increase their resolution. The archive returned by the MOBO/RES is the predicted set of protein structures. Then, a decision-making step is applied to select the final predicted protein structure. To accomplish that, the MUFOLD-CL method [30] is used to cluster the final set.

The MUFOLD-CL works by first estimating potential cluster representatives (center of a cluster) using a structural difference metric. These representatives are then used to cluster the remaining structures, also using this metric. Finally, after all clusters are formed, new representatives are selected for each cluster using a structural similarity metric that is able to describe more accurately the center of a cluster [30]. The representative structure of the largest group is returned as the predicted structure of the proposed method.

⁴<http://old.rosetta.org/>

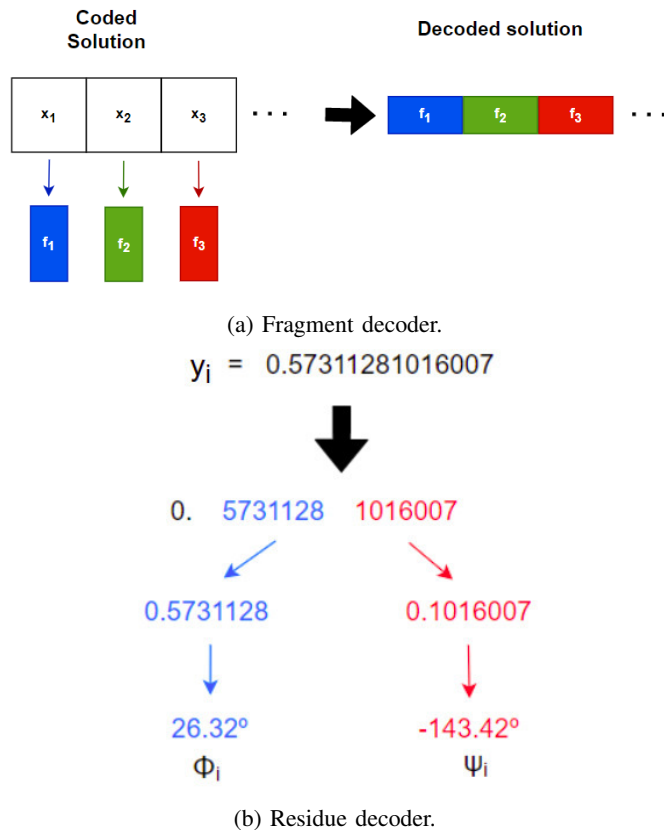


Fig. 3: MOBO decoders

A diagram with a visualization of the complete predictor can be seen in Figure 4. In this diagram, the primary input is the amino acid sequence. The secondary structure, contact map, and fragment information are generated from this sequence. This information is used to feed the optimizer, which starts with MOBO/FRAG method. This method generates an archive of solutions, used as the initial population for the second method, MOBO/RES. The output of the second method is the optimizer output. This output is also an archive of solutions, used as input for the clustering method MUFOLD-CL. The final structure is then selected from the largest cluster found.

V. EXPERIMENTS, RESULTS & ANALYSIS

Table I shows the protein set employed in the experiments. All proteins were taken from the RCSB database⁵, with the exception of proteins T0868, T0900, T0968s1, and T1010, which were taken from the CASP competition⁶. An amount of 20 proteins with different sizes and different types of secondary structures were selected.

The Root Mean Square Deviation (RMSD) metric was utilized to measure the quality of the predicted structures. The RMSD measures the distance of atoms between two superimposed structures, with lower values indicating higher similarity structures [31]. The distance is taken considering

⁵<https://www.rcsb.org/>

⁶<https://predictioncenter.org/>

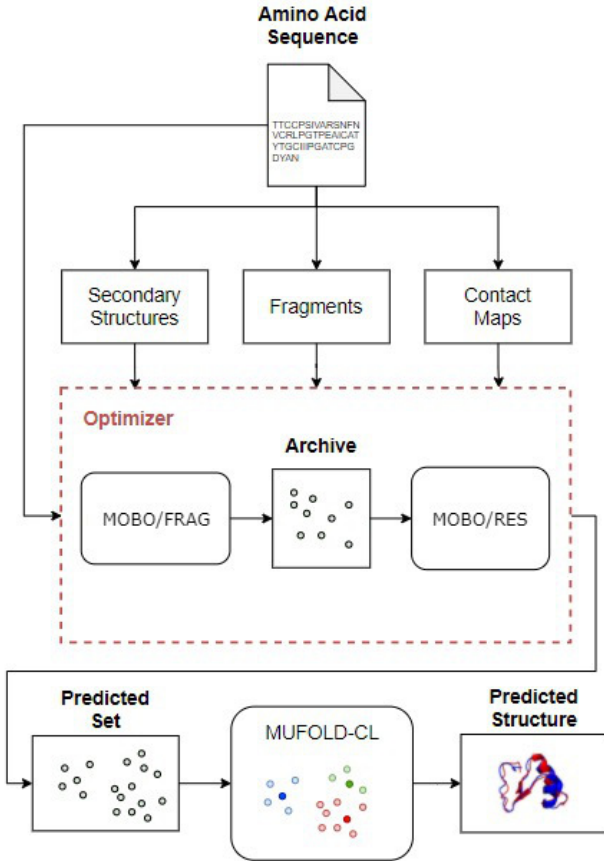


Fig. 4: Predictor

Protein	Size	Class	Protein	Size	Class
1ACW	29	$\alpha\beta$	2KDL	56	α
1DFN	30	β	1BDD	60	α
1ZDD	34	α	1ROP	63	α
1I6C	39	β	1AIL	73	α
2MR9	44	α	1HHP	99	β
2P81	44	α	T0900	106	β
1AB1	46	$\alpha\beta$	T0968s1	119	$\alpha\beta$
1CRN	46	$\alpha\beta$	1ALY	146	β
1ENH	54	α	T0868	161	α
1GB1	56	$\alpha\beta$	T1010	210	β

TABLE I: Proteins utilized in the experiments

the C_α atoms (protein backbone) and can be mathematically defined as:

$$RMSD(\vec{a}, \vec{b}) = \sqrt{\frac{\sum_{i=1}^L d(a_i, b_i)^2}{L}} \quad (10)$$

where L is the number of residues (amino acids), \vec{a} and \vec{b} are two superimposed structures, and $d(a_i, b_i)$ returns the Euclidean distance (in angstroms) between the atoms a_i and b_i . The best and average values (with standard deviation) were measured for all proteins.

All the experiments were executed 20 times for statistical validation. The proposed predictor was implemented using C++17, and experiments were performed on an Ubuntu 18.04 system with an Intel Xeon E7-8860 @ 80x 2.26GHz and 1TB

RAM. The utilized system has a NUMA architecture [32] with four nodes, each node with 20 cores (10 physical and 10 virtual).

The experiments were conducted using the number of iterations $N_{it} = 1000$ and the population size $p = 500$. Both MOBO/FRAG and MOBO/RES use the same values, resulting in 1,000,000 fitness evaluations. However, the MOBO/RES also defines the exploration diversity threshold as $\delta = 0.25$. This definition occurs to limit the exploration of new solutions in the MOBO/RES, whose objective is to refine the solutions found by the MOBO/FRAG. These parameters were empirically defined.

A. Predictor performance analysis

The first experiment compared the results obtained by MOBO against the previous work [12]. The previous algorithm has the same structure as MOBO, including the diversity modifications, but uses static parameters defined before the algorithm execution. The ANOVA test with 95% confidence interval was performed to validate the statistical difference between the result of the MOBO and the previous algorithm. From the results obtained, both algorithms obtained statistically same results in all 20 proteins. As one objective of using online parameter control strategies is to reduce the usability complexity of the algorithm, the MOBO algorithm has an advantage over the previous work with fewer parameters to be adjusted. Therefore, the use of parameter control is validated reducing the number of parameters to be tuned from 6 to 2. The only parameters that still need to be defined are the number of iterations N_{it} and the population size p .

In the following, three types of results are analyzed considering the proposed MOBO predictor:

- Best value of the final archive: $MOBO_{BEST}$;
- Average value of the final archive: $MOBO_{MEAN}$; and
- Value from the structure selected with MUFOLD-CL: $MOBO_{CL}$.

This analysis was performed to verify the overall quality of generated archive and the quality of the solution selected with MUFOLD-CL. Although it is not expected that the decision-maker will always select the best-generated structure, it should at least select a structure with quality higher than the average generated structure.

Comparisons with predictors from the literature were also conducted, which can be seen in Table II. For the Rosetta *de novo* protocol, the set of proteins was predicted locally. The results of the other predictors were extracted directly from their respective works.

The predictor proposed in [21] uses the MUFOLD-CL to select a final solution from the optimized set of structures. In [17], a hierarchical clustering method is used for this purpose. In [18], a single final solution was not selected, using the best solution generated by the algorithms for evaluation instead.

The overall results are shown in Table III. The table compares each protein with the results of all methods. The best and average values from other works were utilized, if available. Missing results are marked with '-' in the table.

The ANOVA test with 95% confidence interval was performed to validate the statistical difference between the result of the proposed predictor MOBO (with MUFOLD-CL) and the other predictors. At the end of Table III, an extra row (B/S/W) was added to summarize the result of this test. B represents the number of proteins where the competing method was statistically better than the proposed predictor, and the W is the number of times where the competing method was statistically worse than the proposed predictor. S is the number of proteins where there was no statistical difference between the results of the competing method and the proposed predictor.

TABLE II: Compared methods

Reference	Algorithm	Function evaluations
[21]	MODE-K	100,000 (10^5)
[18]	NSGA-II GDE3 DEMO	1,000,000 (10^6)
[17]	MOPSO	100,000 (10^5)
Rosetta	<i>de novo</i> protocol	1,000,000 (10^6)
Proposal	MOBO	1,000,000 (10^6)

Considering the best solution found on all executions, the MOBO was able to generate the best solution for almost all proteins except for 1ACW (Rosetta), 1GB1 (Rosetta), 1ZDD (DEMO), and 2P81 (MODE-K). The quality of the solutions found using MUFOLD-CL was not far from the best solution generated by the MOBO. It seems that, on average, the distance between the best solution and the MUFOLD-CL solution is between 1 and 3 Å.

Overall, the solutions selected by the MUFOLD-CL are better than the other methods. The MOBO with MUFOLD-CL selected better solutions most of the time when compared with NSGA2, GDE3, DEMO, and Rosetta.

Considering the best and mean values of the MOBO, the MUFOLD-CL was always worst than the best and almost always similar to the mean. These results make sense, as the MUFOLD-CL is a clustering method and selects as the final result the center of the largest cluster. As this center is similar to all the other structures of the cluster, and the largest cluster is selected, it is expected that the results are close to the mean of the final frontier.

1) *Predicted structures visualization*: The visualization of the predicted structures can be seen in Figure 5. In this figure, the best solution predicted by the MOBO with MUFOLD-CL is in red, while the native structure is in blue. The structures were overlapped using the α -carbons. This overlap displays the quality of the predicted structures.

It is possible to see that the α -helices of the proteins were accurately predicted, in general. This behavior can be seen in the structures of 1AB1, 1BDD, 1CRN, 1ENH, 1GB1, 1ROP, 1ZDD, and 2MR9. However, for some α proteins, the algorithm was unable to generate accurate helices. This can be observed in 2KDL and 2P81, which are small α proteins with poor predictions. This divergence of quality is probably due to the secondary structure and contact map information predicted for these proteins. If the input information has poor

quality, it is expected that the output structure will be equally bad.

It is also possible to see in this visualization that one difficult part is the prediction of β -sheets. These structures are harder to predict than the α -helices, and proteins of the class β are usually the ones with the lowest prediction quality. This difficulty can be observed both in simpler proteins, such as 1DFN, and in the more complex, such as 1ALY and 1HHP.

VI. CONCLUSION AND FUTURE WORK

The PSP problem is one of the most important open problem in biology. As proteins are a core biological macromolecule, understanding their structure and functionality is essential to understanding complex biological processes. For this purpose, computational methods are employed to build and simulate protein structures.

Among possible methods to approach the PSP problem, multi-objective optimization shows great potential. Being able to simultaneously optimize multiple conflicting objectives, these methods can optimize complex mathematical models. Due to this, recent works in the literature have invested multi-objective optimization for the PSP problem.

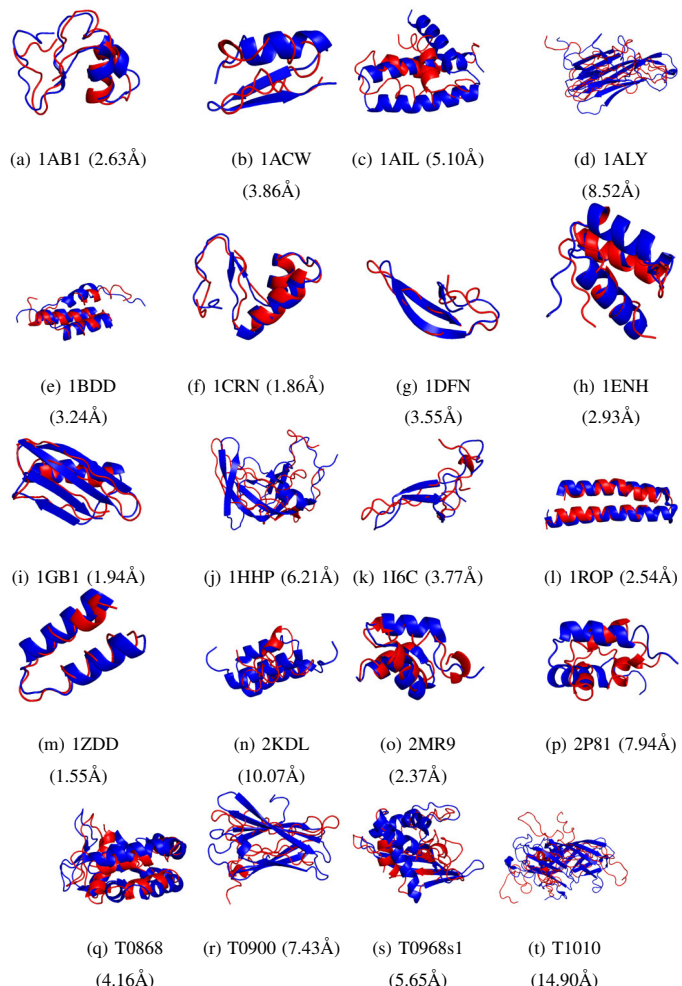


Fig. 5: Overlap of predicted and native figures. The predicted structure is in red and the native structure is in blue.

TABLE III: RMSD results. For each method, the best solution found (f^*), mean value (\bar{x}) and standard deviation (s) are displayed. The best absolute f^* for each protein is set in bold.

Protein		NSGA2	GDE3	DEMO	MOPSO	MODE-K	Rosetta	MOBO _{BEST}	MOBO _{MEAN}	MOBO _{CL}
1AB1	f^*	-	-	-	9.80	7.38	4.90	2.17	4.72	2.63
	\bar{x}	-	-	-	-	-	7.78	2.93	5.81	5.08
	s	-	-	-	-	-	1.20	0.46	0.75	1.61
1ACW	f^*	3.81	3.63	3.82	-	-	1.65	2.71	4.47	3.86
	\bar{x}	6.47	6.56	7.17	-	-	5.48	3.90	5.55	5.29
	s	1.58	1.72	1.81	-	-	1.27	0.58	0.44	0.55
1AIL	f^*	7.07	3.25	3.14	-	-	6.26	3.84	5.96	5.10
	\bar{x}	10.30	6.77	7.40	-	-	9.50	5.72	7.78	7.42
	s	1.38	2.81	2.55	-	-	1.98	0.89	0.81	1.31
1ALY	f^*	-	-	-	-	-	13.62	7.78	14.29	8.52
	\bar{x}	-	-	-	-	-	16.51	11.29	17.29	14.59
	s	-	-	-	-	-	2.14	1.75	1.29	2.93
1BDD	f^*	-	-	-	5.64	4.98	5.27	2.79	3.76	3.24
	\bar{x}	-	-	-	-	-	7.61	3.26	4.15	3.94
	s	-	-	-	-	-	1.60	0.19	0.29	0.59
1CRN	f^*	6.23	5.13	6.32	7.57	-	4.91	1.65	3.53	1.86
	\bar{x}	9.24	9.62	9.31	-	-	7.24	2.52	5.28	4.72
	s	1.50	2.69	2.79	-	-	1.02	0.45	1.04	2.08
1DFN	f^*	-	-	-	-	7.00	5.48	2.68	4.49	3.55
	\bar{x}	-	-	-	-	-	7.10	3.23	5.12	4.72
	s	-	-	-	-	-	0.68	0.34	0.57	1.05
1ENH	f^*	6.84	3.10	4.29	8.92	7.80	3.66	1.94	3.29	2.93
	\bar{x}	10.14	8.09	7.47	-	-	5.00	2.30	3.86	3.67
	s	1.31	2.79	1.67	-	-	1.03	0.26	0.33	0.62
1GB1	f^*	-	-	-	-	-	1.63	1.63	2.53	1.94
	\bar{x}	-	-	-	-	-	2.92	2.45	3.38	2.94
	s	-	-	-	-	-	1.82	0.53	0.63	0.74
1HHP	f^*	-	-	-	-	-	11.32	4.28	8.83	6.21
	\bar{x}	-	-	-	-	-	14.58	8.57	12.80	10.55
	s	-	-	-	-	-	1.28	2.26	1.93	2.63
1I6C	f^*	-	-	-	8.47	7.76	6.84	3.21	4.66	3.77
	\bar{x}	-	-	-	-	-	8.31	3.74	5.37	5.32
	s	-	-	-	-	-	0.76	0.41	0.35	0.76
1ROP	f^*	5.96	2.74	3.04	3.51	3.01	8.34	1.14	2.56	2.54
	\bar{x}	10.86	7.05	6.80	-	-	11.01	1.63	3.56	3.18
	s	1.85	2.07	1.61	-	-	1.52	0.43	0.99	0.52
1ZDD	f^*	3.47	1.79	1.19	2.15	2.50	2.99	1.31	1.54	1.55
	\bar{x}	6.04	4.05	4.01	-	-	5.26	1.58	1.83	1.83
	s	1.29	1.16	1.98	-	-	1.06	0.18	0.22	0.21
2KDL	f^*	-	-	-	10.29	7.72	11.41	6.01	10.57	10.07
	\bar{x}	-	-	-	-	-	12.98	8.55	11.26	11.13
	s	-	-	-	-	-	0.44	0.97	0.37	0.67
2MR9	f^*	6.16	3.00	2.62	-	-	2.21	1.66	2.41	2.37
	\bar{x}	8.29	7.09	7.21	-	-	4.46	1.89	2.61	2.61
	s	1.29	1.87	1.87	-	-	2.62	0.20	0.12	0.18
2P81	f^*	5.85	4.78	5.06	6.28	4.76	5.56	5.89	8.33	7.94
	\bar{x}	8.94	7.10	7.72	-	-	7.97	7.12	9.08	8.94
	s	1.30	1.34	1.44	-	-	1.34	0.67	0.37	0.49
T0868	f^*	-	-	-	-	-	13.06	3.38	6.00	4.16
	\bar{x}	-	-	-	-	-	15.26	5.20	8.82	6.90
	s	-	-	-	-	-	1.41	1.15	1.46	1.63
T0900	f^*	-	-	-	-	-	12.71	5.93	10.72	7.43
	\bar{x}	-	-	-	-	-	15.07	7.39	12.21	9.40
	s	-	-	-	-	-	0.88	0.74	0.66	1.33
T0968S1	f^*	-	-	-	-	-	12.79	4.85	8.47	5.65
	\bar{x}	-	-	-	-	-	15.78	6.50	11.43	8.93
	s	-	-	-	-	-	1.67	1.02	1.49	2.00
T1010	f^*	-	-	-	-	-	18.37	12.86	20.68	14.90
	\bar{x}	-	-	-	-	-	21.13	14.36	22.50	19.46
	s	-	-	-	-	-	1.43	0.88	0.97	3.53
B/S/W		0/1/7	1/1/6	1/1/6	-	-	1/3/16	20/0/0	0/14/6	-

The results obtained by the proposed approach demonstrated that the predictor with the decision-making step is highly competitive with other works from the literature. Although the predictor is consistently able to predict α -helices, the

prediction of β -sheet is unstable, with proteins of the β class having considerably lower quality when compared with proteins of the α class.

It is possible to define some improvements in the proposed

work. Multiple predictors could be used to reduce the inaccuracy of predicted information in the optimization. By using the information of different predictors, it is possible to reduce bias and combine the best information from each source. This combination of multiple data sources may increase the effectiveness of secondary structures and contact maps in the prediction of tertiary structures.

Also, the prediction of β should be more thoroughly researched. Other types of information could be added to the optimization model with the objective of increasing the quality of β -sheets in protein predictions. By focusing on this weak point, the overall effectiveness of the predictor should increase not only for β proteins but also for all proteins in general.

Another line of work is to further improve the search algorithm itself. This work presents a simple online parameter control that is able to select reasonable values. However, more complex techniques could be employed, such as the use of fuzzy systems to incorporate expert information about the problem that is being optimized. Also, local search could be implemented to further specialize the proposed algorithm for the optimization of protein structures.

REFERENCES

- [1] R. Garret and C. Grisham, "Biochemistry 4ed," *University of Virginia, Boston, MA*, 2010.
- [2] K. A. Dill, S. B. Ozkan, M. S. Shell, and T. R. Weikl, "The protein folding problem," *Annu. Rev. Biophys.*, vol. 37, pp. 289–316, 2008.
- [3] J. Gu and P. E. Bourne, *Structural bioinformatics*. Hoboken: John Wiley & Sons, 2009, vol. 44.
- [4] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Židek, A. Potapenko *et al.*, "Highly accurate protein structure prediction with alphafold," *Nature*, vol. 596, no. 7873, pp. 583–589, 2021.
- [5] A. David, S. Islam, E. Tankhilevich, and M. J. Sternberg, "The alphafold database of protein structures: A biologist's guide," *Journal of Molecular Biology*, vol. 434, no. 2, p. 167336, 2022.
- [6] M. Dorn, M. B. e Silva, L. S. Buriol, and L. C. Lamb, "Three-dimensional protein structure prediction: Methods and computational strategies," *Computational biology and chemistry*, vol. 53, pp. 251–276, 2014.
- [7] A. E. Márquez-Chamorro, G. Asencio-Cortés, C. E. Santiesteban-Toca, and J. S. Aguilar-Ruiz, "Soft computing methods for the prediction of protein tertiary structures: A survey," *Applied Soft Computing*, vol. 35, pp. 398–410, 2015.
- [8] R. S. Silva and R. S. Parpinelli, "A self-adaptive differential evolution with fragment insertion for the protein structure prediction problem," in *International Workshop on Hybrid Metaheuristics*. Springer, 2019, pp. 136–149.
- [9] N. N. Will and R. S. Parpinelli, "Comparing best and quota fragment picker protocols applied to protein structure prediction." in *HIS*, 2020, pp. 669–678.
- [10] V. Cutello, G. Narzisi, and G. Nicosia, "A multi-objective evolutionary approach to the protein structure prediction problem," *Journal of The Royal Society Interface*, vol. 3, no. 6, pp. 139–151, 2006.
- [11] D. Kalyanmoy and K. Deb, "Multi-objective optimization using evolutionary algorithms," *West Sussex, England: John Wiley*, 2001.
- [12] F. Marchi and R. S. Parpinelli, "A multi-objective approach to the protein structure prediction problem using the biased random-key genetic algorithm," in *2021 IEEE Congress on Evolutionary Computation (CEC)*. New York: IEEE, 2021, pp. 1070–1077.
- [13] J. F. Gonçalves and M. G. Resende, "Biased random-key genetic algorithms for combinatorial optimization," *Journal of Heuristics*, vol. 17, no. 5, pp. 487–525, 2011.
- [14] S. M. Venske, R. A. Gonçalves, E. M. Benelli, and M. R. Delgado, "Ade-mo/d: An adaptive differential evolution for protein structure prediction problem," *Expert Systems with Applications*, vol. 56, pp. 209–226, 2016.
- [15] S. Gao, S. Song, J. Cheng, Y. Todo, and M. Zhou, "Incorporation of solvent effect into multi-objective evolutionary algorithm for improved protein structure prediction," *IEEE/ACM transactions on computational biology and bioinformatics*, vol. 15, no. 4, pp. 1365–1378, 2017.
- [16] S. Song, S. Gao, X. Chen, D. Jia, X. Qian, and Y. Todo, "Aimoes: Archive information assisted multi-objective evolutionary strategy for ab initio protein structure prediction," *Knowledge-Based Systems*, vol. 146, pp. 58–72, 2018.
- [17] S. Song, J. Ji, X. Chen, S. Gao, Z. Tang, and Y. Todo, "Adoption of an improved pso to explore a compound multi-objective energy function in protein structure prediction," *Applied Soft Computing*, vol. 72, pp. 539–551, 2018.
- [18] P. H. Narloch, M. J. Krause, and M. Dorn, "Multi-objective differential evolution algorithms for the protein structure prediction problem," in *2020 IEEE Congress on Evolutionary Computation (CEC)*. New York: IEEE, 2020, pp. 1–8.
- [19] G. K. Rocha, K. B. Dos Santos, J. S. Angelo, F. L. Custodio, H. J. Barbosa, and L. E. Dardenne, "Inserting co-evolution information from contact maps into a multiobjective genetic algorithm for protein structure prediction," in *2018 IEEE Congress on Evolutionary Computation (CEC)*. New York: IEEE, 2018, pp. 1–8.
- [20] A. B. Zaman, P. V. Parthasarathy, and A. Shehu, "Using sequence-predicted contacts to guide template-free protein structure prediction," in *Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, ser. BCB '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 154–160. [Online]. Available: <https://doi.org/10.1145/3307339.3342175>
- [21] X. Chen, S. Song, J. Ji, Z. Tang, and Y. Todo, "Incorporating a multiobjective knowledge-based energy function into differential evolution for protein structure prediction," *Information Sciences*, vol. 540, pp. 69–88, 2020.
- [22] L. de Lima Corrêa and M. Dorn, "A multi-objective swarm-based algorithm for the prediction of protein structures," in *International Conference on Computational Science*. Cham: Springer, 2019, pp. 101–115.
- [23] J. C. C. Tudela and J. O. Lopera, "Parallel protein structure prediction by multiobjective optimization," in *2009 17th Euromicro International Conference on Parallel, Distributed and Network-based Processing*. New York: IEEE, 2009, pp. 268–275.
- [24] C. A. Rohl, C. E. Strauss, K. M. Misura, and D. Baker, "Protein structure prediction using rosetta," *Methods in enzymology*, vol. 383, pp. 66–93, 2004.
- [25] D. T. Jones, "Protein secondary structure prediction based on position-specific scoring matrices," *Journal of molecular biology*, vol. 292, no. 2, pp. 195–202, 1999.
- [26] W. Kabsch and C. Sander, "Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features," *Biopolymers: Original Research on Biomolecules*, vol. 22, no. 12, pp. 2577–2637, 1983.
- [27] J. Ma, S. Wang, Z. Wang, and J. Xu, "Protein contact prediction by integrating joint evolutionary coupling analysis and supervised learning," *Bioinformatics*, vol. 31, no. 21, pp. 3506–3513, 2015.
- [28] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197, 2002.
- [29] R. Parpinelli, G. Plichoski, R. Silva, and P. Narloch, "A review of techniques for online control of parameters in swarm intelligence and evolutionary computation algorithms," *International Journal of Bio-Inspired Computation*, vol. 13, pp. 1–20, 2019.
- [30] J. Zhang and D. Xu, "Fast algorithm for population-based protein structural model analysis," *Proteomics*, vol. 13, no. 2, pp. 221–229, 2013.
- [31] V. N. Maiorov and G. M. Crippen, "Significance of root-mean-square deviation in comparing three-dimensional structures of globular proteins," *Journal of molecular biology*, vol. 235, no. 2, pp. 625–634, 1994.
- [32] H. El-Rewini and M. Abd-El-Barr, *Advanced computer architecture and parallel processing*. Hoboken, New Jersey: John Wiley & Sons, 2005, vol. 42.

Team Orienteering Problem with Time Windows and Variable Profit

Eliseo Marzal, Laura Sebastia
 Valencian Research Institute for Artificial Intelligence
 Universitat Politècnica de València
 Camino de Vera s/n - Valencia (Spain)
 Email: {emarzal, lsebastia}@dsic.upv.es

Abstract—This paper introduces the **Team Orienteering Problem with Time Windows and Variable Profits (TOPWPVP)**, which is a variant of the classical **Orienteering Problem (OP)** and an **NP-hard optimization problem**. It consists in determining the optimal route for a vehicle to traverse to deliver to a given set of nodes (customers), where each node has a predefined time window in which the service must start (in case this node is visited), and the vehicle may spend an amount of time given by a predefined interval so that the profit collected at this node depends on the time spent. We first propose a mathematical model for the **TOPTWVP**, then propose an algorithm based on **Iterated Local Search** to solve modified benchmark instances. The results show that our approach can solve difficult instances with good quality.

I. INTRODUCTION

THE Orienteering Problem (OP) [1] is a combinatorial optimization and integer programming problem whose goal is to obtain the optimal route for a vehicle to traverse to deliver to a given set of customers. The objective is to maximize the total score collected from visited (selected) nodes. The Team OP (TOP) [2] is one of the most studied variants of the OP, where the route for several vehicles must be computed. When a time window is established for each node so that the service at a particular node has to start within a predefined time window, a new variant, called (Team) Orienteering Problem with Time Windows ((T)OPTW) [3], is defined. Another variant is the orienteering problem with variable profit (OPVP) [4] and its generalization, the team orienteering problem with variable profit (TOPVP) [5]. In this case, the profit collected at each node depends on the number of times a node is visited or on the continuous amount of time spent at a given node.

In this paper, we combine the TOPTW and the TOPVP to come up with the **Team Orienteering Problem with Time Windows and Variable Profit (TOPTWVP)**. Specifically, each node has a predefined time window in which the service must start (in case this node is visited) and the vehicle may spend an amount of time given by a predefined interval so that the profit collected at this node depends on the time spent.

This extension of the TOPTW and the TOPVP is especially interesting to applications of the vehicle routing problems

This work has been partially supported by the Spanish MINECO project TIN2017-88476-C2-1-R, the AVI project SmartTur+ECO INNEST/2021/233 and the EU project TAILOR 952215.

such as the routing of a reconnaissance vehicle [6], which has the option of staying longer at a location to gather more information and the route planning may depend on time windows when it is safer to perform this reconnaissance task; or the the Tourist Trip Design Problems [7], where a longer stay at a location may provide a higher satisfaction (profit) to the visitor and the visits must be scheduled taking into account the opening times of attractions. For example, the work in [8], [9] and [10] use a modified version of the TOPTWVP which allow the tourist to define some travel-style preferences, such as if they prefer to visit a few or many places or if they prefer to enjoy some free time during the trip. In order to consider these travel-style preferences, each visit is assigned a duration interval so that each visit finally takes the most appropriate time to fit into the tourist preferences.

In general, TOPTWVP tries to model a more realistic situation, where locations can only be visited within specific time windows and where staying longer (also within a given interval) may report a higher profit.

The main contributions of this paper are the following:

- We introduce a mathematical model for the TOPTWVP.
- Due to the limitation of solving large instances, we then propose a heuristic based on Iterated Local Search (ILS).
- We perform experiments with the ILS algorithm on instances for the Solomon benchmark.
- We compare the solutions with respect to several Key Performance Indicators (KPIs): score, number of visits, waiting time (idle time in the route), and execution time.

The remainder of this paper is as follows. The problem description and the mathematical model is detailed in Section II. Section III describes the proposed algorithm based on ILS. Section IV reports numerical experiments performed on benchmark instances. Finally, in Section V, we summarize the main achievements and future works.

II. TOP WITH TIME WINDOWS AND VARIABLE PROFIT

The input values in the **TOPTWVP** are the following:

- a set of nodes $N = \{1, \dots, |N|\}$, where nodes 1 and $|N|$ represent the start and end nodes
- a number of routes M and total time budget per route T_{max}
- For each node $i \in N$:

- a travel time t_{ij} from node i to j , known for all vertices
- a non-negative profit P_i , i.e. the maximum profit collected when visiting a node; P_1 and $P_{|N|}$ are 0
- a time window $[O_i, C_i]$, which indicates that a visit to a node can only start during this time window
- a service time interval $[minD_i, maxD_i]$, i.e. the minimum and maximum amount of time that can be spent at a node

The goal of the TOPTWVP is to determine M routes, each limited by T_{max} , that visit a subset of N within the respective time windows, during a service time in the interval duration and that maximize the total collected profit. The TOPTWVP can be formulated as an integer programming model with the following decision variables:

- $x_{ijm} = 1$, if in route m , a visit to node i is followed by a visit to node j ; 0, otherwise
- $y_{im} = 1$, if node i is visited in route m ; 0, otherwise.
- s_{im} is the start of the service at node i in route m
- d_{im} is the service time (duration) at node i in route m

And the following constraints:

$$\sum_{m=1}^M \sum_{j=2}^{|N|} x_{1jm} = \sum_{m=1}^M \sum_{i=1}^{|N|-1} x_{i|N|m} = M \quad (1)$$

$$\sum_{m=1}^M y_{km} \leq 1; \forall k = 2, \dots, (|N| - 1) \quad (2)$$

$$\sum_{i=1}^{|N|-1} x_{ikm} = \sum_{j=2}^{|N|} x_{kjm} = y_{km}; \quad \forall k = 2, \dots, (|N| - 1); \forall m = 1, \dots, M \quad (3)$$

$$O_i \leq s_{im} \leq C_i; \forall i = 1, \dots, |N|; \forall m = 1, \dots, M \quad (4)$$

$$s_{im} + d_{im} + t_{ij} - s_{jm} \leq L(1 - x_{ijm}); \quad \forall i, j = 1, \dots, |N|; \forall m = 1, \dots, M \quad (5)$$

$$minD_i \leq d_{im} \leq maxD_i; \forall m = 1, \dots, M \quad (6)$$

Constraints 1 establish that each route starts from node 1 and ends in $|N|$. Constraints 2 ensure that each node can only be visited at most once in all routes. Constraints 3 ensure the connectivity of each route. Constraints 4 force that the service starts within the time window for each route. Constraints 5 ensure the timeline of the route (L is a large constant, that can be equal to T_{max}), explicitly considering the variable duration of the service. Constraint 6 guarantees that the service time is within the corresponding interval of duration.

$$Maximize \sum_{m=1}^M \sum_{i=2}^{|N|-1} P_i d_{im} y_{im} \quad (7)$$

The maximization function 7 establishes that the profit of the final plan not only depends on whether a node is visited or not but also it depends on how long a node is visited. Without loss of generality, in this definition of the TOPTWVP, we assume that each time unit spent at the node collects P_i . Therefore, the total score of a visited node i results from the product of the profit P_i and the time spent at a node d_{im} (in a route m).

Since the TOPTW and TOPVP are NP-hard, the TOPTWVP is also NP-hard. Both dealing with time windows and with an interval of possible service times makes this problem harder to solve. We have performed some experiments (not shown in this paper due to space restrictions) that prove that our implementation of this mathematical model is able to solve only small instances. Moreover, as some works referred to in this paper also state, in general, variations of TOP problems with a significant number of nodes are solved with heuristic approaches, and exact algorithms are only feasible for problems with a small number of nodes. For this reason, the following section introduces a heuristic approach to solve the TOPTWVP.

III. HEURISTIC APPROACH TO TOPTWVP

The more successful heuristic approaches for both the TOPTW and TOPVP are based on the ILS algorithm ([11],[12],[5]). Specifically, in this work, we take the ILS implementation for solving the TOPTW given in [11] and [12] and adapt some steps in order to consider the variable profit.

The general scheme of the ILS is shown in Algorithm 1. In a nutshell, this algorithm constructs an initial feasible solution (Construction), which is further improved by Iterated Local Search (ILS). ILS comprises the components LocalSearch, Perturbation and AcceptanceCriterion. We refer the reader to [11] and [12] for further details in the ILS implementation. In this section, we only focus on the modifications introduced to solve the specific TOPTWVP.

Apart from the input data and decision variables detailed in the previous section, we also keep the following data for each node, to reduce the time devoted to checking the feasibility of a route, thus making the algorithm more efficient [11]:

- $arrive_{im}$: arriving time to a node. If $arrive_{im} \in [O_i, C_i]$, then the arriving time and the start time s_{im} take the same value.
- $wait_{im}$: waiting time to visit a node because the arriving time $arrive_{im}$ is previous to the opening time O_i .
- $MaxShift_{im}$: maximum time that a visit can be delayed so that the route is still feasible, i.e. it keeps track of how much a certain visit can be shifted in time, without violating any time window in the route.

Regarding the Construction process, the first solution only contains the start node or depot, as initial and final nodes. Then, a set F with all feasible candidate nodes that can be visited is generated. All possibilities of inserting an unscheduled node in position p of route m are examined. To check the feasibility of inserting node j between nodes i and k , we

Algorithm 1 ILS(N, M)

```

1:  $S_0 \leftarrow \text{Construction}(N, M)$ 
2:  $S_0 \leftarrow \text{LocalSearch}(S_0, N^*, N', M)$ 
3:  $S^* \leftarrow S_0$ 
4:  $\text{NoImprovement} \leftarrow 0$ 
5: while TimeLimit has not been reached do
6:    $S_0 \leftarrow \text{Perturbation}(S_0, N^*, N', M)$ 
7:    $S_0 \leftarrow \text{LocalSearch}(S_0, N^*, N', M)$ 
8:   if  $S_0$  better than  $S^*$  then
9:      $S^* \leftarrow S_0$ 
10:     $\text{NoImprovement} \leftarrow 0$ 
11:   else
12:      $\text{NoImprovement} \leftarrow \text{NoImprovement} + 1$ 
13:   end if
14:   if  $(\text{NoImprovement} + 1) \bmod \text{ThresholdImpr} = 0$ 
15:     then
16:        $S_0 \leftarrow S^*$ 
17:     end if
18: end while
19: return  $S^*$ 

```

compute Shift_{jm} , which is time added or reduced when a new node is inserted or removed in a route:

$$\text{Shift}_{jm} = t_{ij} + \text{wait}_{jm} + d_{jm} + t_{jk} - t_{ik}$$

where: $\text{wait}_{jm} = \max(0, O_j - \text{arrive}_{jm})$ and $\text{arrive}_{jm} = s_{im} + d_{im} + t_{ij}$. Therefore, it is feasible to insert a node j between nodes i and k in route m if $\text{Shift}_{jm} \leq \text{wait}_{km} + \text{MaxShift}_{km}$. In the TOPTWVP, the service time d_{jm} is not a value known a priori. Therefore, to perform the calculations above, it is necessary to compute an estimate of the service time, which is based on using all the available time without causing infeasibility in the route:

$$d_{jm} = \max(\min D_j, \min(\max D_j, MS')), \text{ where :}$$

$$MS' = \text{wait}_{km} + \text{MaxShift}_{km} - t_{ij} - \text{wait}_{jm} - t_{jk} + t_{ik}$$

The set F contains all the feasible candidate nodes; one of these nodes i is selected according to the attractiveness of the insertion, which is computed as P_i^2 / Shift_i . Once a node i is selected, it is inserted in the corresponding route and position, and S_0 , N' , and N^* are updated accordingly. The service time d_{im} for the selected node is set to the estimate computed above. Consequently, the value of arrive_{jm} , wait_{jm} , s_{jm} , Shift_{jm} and MaxShift_{jm} of the subsequent nodes is also updated. Additionally, MaxShift_{jm} of the previous nodes is also recomputed. The construction process of the first solution is terminated when $F = \emptyset$, that is, when no more feasible candidate nodes are found. This solution will be further improved by the ILS.

Our LocalSearch procedure is composed by the same six local search moves described in [12], and we add a new move that modifies the service time of some nodes. These seven local search moves are applied in three stages. The first stage contains the moves Swap1, Swap2, 2-Opt and Move which

Algorithm 2 IncreaseServiceTime(S_0, M)

```

1: for each route  $m \in M$  do
2:   while  $\exists i : \text{MaxShift}_{im} > 0$  do
3:      $G = \{i : \text{MaxShift}_{im} > 0\}$ 
4:     for  $i \in G$  do
5:        $\Delta_{im} \leftarrow \min(\text{MaxShift}_{im}, \max D_i - d_{im})$ 
6:     end for
7:      $n^* = \text{argmax}_{i \in G} (P_i * \Delta_{im})$ 
8:      $d_{n^*m} \leftarrow d_{n^*m} + \Delta_{n^*m}$ 
9:     for each node  $j$  in route  $m: s_{jm} > s_{n^*m}$  do
10:      Update( $\text{arrive}_{jm}, s_{jm}, \text{Shift}_{jm}, \text{MaxShift}_{jm}$ )
11:    end for
12:    for each node  $k$  in route  $m: s_{km} \leq s_{n^*m}$  do
13:      Update( $\text{MaxShift}_{km}$ )
14:    end for
15:  end while
16: end for
17: return  $S_0$ 

```

restructure the current solution trying to decrease the travel total cost, thus increasing the unused time budget.

The second stage, named IncreaseServiceTime, consists of a new move aiming to enlarge some nodes' service time, thus increasing the total score. Specifically, Algorithm 2 is applied. Once the first stage has been executed, and at least one move has been applied, a restructured solution with an increased unused time budget is obtained; that is, some idle time can be used to increase the service time of specific nodes. Therefore, our aim at this moment is to find a node whose service time can be enlarged and, consequently, the solution profit improves. Algorithm 2 describes this process. For each route, we build the set G with the nodes whose service time can be increased (line 3) because they have a positive MaxShift. Then, this increment in the service time is computed for each node in G (lines 4-6), and the node n^* with the highest increment in the score is selected (line 7). The new service time d_{n^*m} is computed and, hence, the value of arrive_{jm} , wait_{jm} , s_{jm} , Shift_{jm} and MaxShift_{jm} of the subsequent nodes and the value of MaxShift_{jm} of the previous nodes are also recomputed.

The third stage in the LocalSearch procedure contains the moves Insert and Replace, focused on increasing the profit. Finally, for both the Perturbation and AcceptanceCriterion components of ILS, in our implementation, we follow the same scheme as [12].

IV. EXPERIMENTS

This section shows the experiments performed in order to validate our algorithm to solve the TOPTWVP problem for $m = 1$ and $m = 2$. A set of 56 TOPTW Solomon (set c) instances of vehicle routing problems with time windows were selected (<https://unicen.smu.edu.sg/oplib-orienteeering-problem-library>) and adapted to the TOPTWVP by adding the interval of service time; in particular, \min_D was set to the service time indicated in the original problems,

whereas max_D is set to $min_D + 30$. For comparison, we have executed a TOPTW implementation with the original Solomon instances (where the service time coincides with min_D), denoted as **Min** and with these same instances where the service time is set to max_D , denoted as **Max**. Table I show the results for score, number of visited nodes, waiting time and execution time for one and two routes, when executing the two baselines **Min** and **Max** and our implementation **Var**; columns under **AVG** show the average value among the ten runs of our ILS solving the TOPTWVP for each instance and columns under **MAX** show the maximum value of these ten runs.

For the **score**, in the case of the **AVG** results, **Max** obtains a higher score in 21 instances for one route and 25 for two routes. On average, **Var** obtains a higher score than **Max** with one route, whereas **Max** gets a better score than **Var** with two routes (although the difference is smaller). Regarding **MAX** results, the solution with **Var** is the one that provides the highest score in many problems, except in 16 instances out of 56 for one route and 24 for two routes, where **Max** obtains a higher score. **Min** is always quite far from the other two approaches, as expected.

Regarding the **number of visits**, the **Var** approach (almost) always include few more visits in the routes for instances solved with one and two routes. In the case of **Max**, fewer nodes tend to be incorporated as they have a longer duration.

The **waiting time** is defined as the sum of the waiting time of all visits, that is, the idle time in the routes. **Var** obtains routes with less waiting time than **Max** for one route: for **AVG** results, **Var** obtains routes with less waiting time in 26 instances versus 14 instances where **Max** obtains routes with less waiting time. However, for two routes, the results are more similar (26 versus 24 instances). Looking into the individual solutions, we have observed that, in many cases, this worsening of the waiting time is due to an improvement in the score, because the maximization function only considers the score and not the waiting time. That is, it is prioritized to include shorter visits that provide a higher score, although it may imply having idle time in the route waiting for the start time of the time window.

The **execution time** corresponds to the time required to find the best solution, although each execution takes all the given time. It can be observed that **Var** always needs longer to obtain a solution since an additional dimension is being introduced by having to work with an interval for the duration of the service time. This makes the problem more complicated, and it is reflected in the time needed to find a solution.

V. CONCLUSIONS

This paper has introduced the Team OP with Time Windows and Variable Profits where each node has a predefined time window in which the service must start (if visited), and the vehicle may spend an amount of time given by a predefined interval so that the profit collected at this node depends on the time spent. We have proposed a mathematical model for solving the TOPTWVP, and an ILS algorithm to solve modified benchmark instances. The results show that our

	Score		Num. Visits		Waiting Time		Exec. Time	
	m=1	m=2	m=1	m=2	m=1	m=2	m=1	m=2
AVG								
Min	21039,39	34748,93	24,34	47,81	6,77	9,04	485,39	615,44
Max	26523,71	46150,14	11,69	23,29	5,97	6,75	269,83	513,86
Var	26813,11	46031,54	12,63	25,44	3,60	7,43	701,07	792,21
MAX								
Min	21316,07	35104,82	24,34	48,58	4,31	6,88	562,30	679,10
Max	26907,86	46726,43	11,69	23,80	4,64	5,20	339,02	547,01
Var	27330,80	46948,29	12,71	25,75	2,94	7,26	715,62	760,83

TABLE I: Summary of score, number of visits, waiting time and execution time of the ILS algorithm with **Min**, **Max** and **Var** for $m = 1$ and $m = 2$

approach can solve difficult instances with good quality. When compared with two baselines **Min** and **Max**, our approach is, in general, obtaining better scores and also better values in other KPIs, such as the number of visits and the waiting time.

As future work, we are developing new procedures to handle the variable profit with the aim of improving the execution time. Moreover, we are working on a new variant of TOPTWVP that allows defining a custom waiting time so that it can also be considered in the optimization function.

REFERENCES

- [1] P. Vansteenwegen, W. Souffriau, and D. Van Oudheusden, "The orienteering problem: A survey," *European Journal of Operational Research*, vol. 209, no. 1, pp. 1–10, 2011. doi: 10.1016/j.ejor.2010.03.045
- [2] I.-M. Chao, B. L. Golden, and E. A. Wasil, "The team orienteering problem," *European journal of operational research*, vol. 88, no. 3, pp. 464–474, 1996. doi: 10.1016/0377-2217(94)00289-4
- [3] S. Boussier, D. Feillet, and M. Gendreau, "An exact algorithm for team orienteering problems," *4or*, vol. 5, no. 3, pp. 211–230, 2007. doi: 10.1007/s10288-006-0009-1
- [4] G. Erdogan and G. Laporte, "The orienteering problem with variable profits," *Networks*, vol. 61, pp. 104–116, 2013. doi: 10.1002/net.21496
- [5] A. Gunawan, K. M. Ng, G. Kendall, and J. Lai, "An iterated local search algorithm for the team orienteering problem with variable profits," *Engineering Optimization*, vol. 50, no. 7, pp. 1148–1163, 2018. doi: 10.1080/0305215X.2017.1417398
- [6] F. Mufalli, R. Batta, and R. Nagi, "Simultaneous sensor selection and routing of unmanned aerial vehicles for complex mission plans," *Computers & Operations Research*, vol. 39, no. 11, pp. 2787–2799, 2012. doi: 10.1016/j.cor.2012.02.010
- [7] P. Vansteenwegen and D. Van Oudheusden, "The mobile tourist guide: an or opportunity," *OR insight*, vol. 20, no. 3, pp. 21–27, 2007. doi: 10.1057/ori.2007.17
- [8] J. Ibáñez, L. Sebastia, and E. Onaindia, "Planning tourist agendas for different travel styles," in *Proceedings of the Twenty-second European Conference on Artificial Intelligence*. IOS Press, 2016. doi: 10.3233/978-1-61499-672-9-1818 pp. 1818–1823.
- [9] J. Ibáñez-Ruiz, L. Sebastián, and E. Onaindia, "Evaluating the quality of tourist agendas customized to different travel styles," *arXiv preprint arXiv:1706.05518*, 2017.
- [10] L. Sebastia and E. Marzal, "Extensions of the tourist travel design problem for different travel styles," *Procedia Computer Science*, vol. 176, pp. 339–348, 2020. doi: 10.1016/j.procs.2020.08.036
- [11] P. Vansteenwegen, W. Souffriau, G. V. Berghe, and D. Van Oudheusden, "Iterated local search for the team orienteering problem with time windows," *Computers & Operations Research*, vol. 36, no. 12, pp. 3281–3290, 2009. doi: 10.1016/j.cor.2009.03.008
- [12] A. Gunawan, H. C. Lau, and K. Lu, "An iterated local search algorithm for solving the orienteering problem with time windows," in *Evolutionary Computation in Combinatorial Optimization: 15th European Conference, EvoCOP 2015, Copenhagen, Denmark, April 8-10, 2015, Proceedings*, 2015. doi: 10.1007/978-3-319-16468-7_6 pp. 61–73.

Subcaterpillar Isomorphism: Subtree Isomorphism Restricted Pattern Trees To Caterpillars

Tomoya Miyazaki

Graduate School of Computer Science and Systems Engineering
 Kyushu Institute of Technology
 Kawazu 680-4, Iizuka 820-8502, Japan
 Email: miyazaki.tomoya481@mail.kyutech.jp

Kouichi Hirata

Department of Artificial Intelligence
 Kyushu Institute of Technology
 Kawazu 680-4, Iizuka 820-8502, Japan
 Email: hirata@ai.kyutech.ac.jp

Abstract—In this paper, we investigate a *subcaterpillar isomorphism* that is a problem, for a rooted labeled caterpillar P and a rooted labeled tree T , of determining whether or not there exists a subtree in T which is isomorphic to P . Then, we design two algorithms to solve the subcaterpillar isomorphism for a caterpillar P and a tree T in (i) $O(p + tDh\sigma)$ time and $O(Dh)$ space and in (ii) $O(p + tD\sigma)$ time and $O(D(h + H))$ space, respectively. Here, p is the number of vertices in P , t is the number of vertices in T , h is the height of P , H is the height of T , σ is the number of alphabets for labels and D is the degree of T . Furthermore, we give experimental results of the two algorithms for artificial data and real data.

I. INTRODUCTION

THE PATTERN matching for tree-structured data such as HTML and XML documents for web mining or DNA and glycan data for bioinformatics is one of the fundamental tasks for information retrieval or query processing. As such pattern matching for *rooted labeled unordered trees* (a tree, for short), a *subtree isomorphism* is the problem of determining, for a *pattern tree* P and a *text tree* T , whether or not there exists a subtree of T which is isomorphic to P . It is known that the subtree isomorphism can be solved in $O(p^{1.5}t/\log p)$ time [10], where p is the number of vertices in P and t is the number of vertices in T . On the other hand, it cannot be solved in $O(t^{2-\varepsilon})$ time for every ε ($0 < \varepsilon < 1$) under SETH [1].

In this paper, we focus on *subcaterpillar isomorphism* that is a subtree isomorphism when P is a *rooted labeled caterpillar* (a *caterpillar*, for short) (cf., [3]). The caterpillar is an unordered tree transformed to a rooted path after removing all the leaves in it. The caterpillar provides the structural restriction of the tractability of computing the *edit distance* [8] and *inclusion problem* [7] for unordered trees.

It is known that the problem of computing the edit distance between unordered trees is MAX SNP-hard [11]. This statement also holds even if two trees are binary, the maximum height is at most 3 or the cost function is the unit cost function [2], [4]. On the other hand, we can compute the edit distance between caterpillars in $O(n + H^2\sigma^3)$ time in the general cost function and $O(n + H^2\sigma)$ time under the unit cost function, where n is the total number of vertices of the two caterpillars, H is the maximum height of the two caterpillars and σ is

the number of alphabets for labels in the two caterpillars [8]¹.

It is known that the inclusion problem of determining whether or not a text tree T achieves to a pattern tree P by deleting vertices in T is NP-complete [6]. This statement also holds even if P is a caterpillar [6]. On the other hand, if both P and T are caterpillars, then we can solve the inclusion problem in $O(p + t + (h + H)\sigma)$ time, where h is the height of P and H is the height of T [7]².

In this paper, we design two algorithms to solve the subcaterpillar isomorphism in (i) $O(p + tDh\sigma)$ time and $O(Dh)$ space and (ii) $O(p + tD\sigma)$ time and $O(D(h + H))$ space, respectively. Here, D is the degree of T . Since there may exist many matching positions that match P in T when P is much smaller than T , the above algorithms also output all of such positions. Hence, under the assumption that $p < t$, $h \ll t$ and $h < H$, the algorithm (i) runs in $O(tD\sigma)$ time and $O(Dh)$ space and the algorithm (ii) runs in $O(tD\sigma)$ time and $O(DH)$ space.

Note that both algorithms do not use the maximum cardinality matching algorithm for bipartite graphs [5], which is essential for the subtree isomorphism algorithm [10]. Also we cannot apply the proof of the SETH-hardness in [1] when a pattern tree P is a caterpillar.

Furthermore, by implementing the algorithms (i) and (ii), we give experimental results of the two algorithms for artificial data and real data. Then, we confirm that, whereas the algorithm (ii) is faster than the algorithm (i) as same as the theoretical results for artificial data of which number of matching positions is large, the algorithm (i) is faster than the algorithm (ii) for real data.

II. PRELIMINARIES

A *tree* is a connected graph without cycles. For a tree $T = (V, E)$, we denote V and E by $V(T)$ and $E(T)$. We sometimes

¹The time complexity represented in [8] is $O(H^2\lambda^3)$ time and $O(H^2\lambda)$ time, where λ is the maximum number of leaves in the two caterpillars. Since $O(\lambda^3)$ and $O(\lambda)$ in them are corresponding to the time complexity of computing the multiset edit distances under the general and the unit cost functions (cf. [9]), we can replace λ with σ , by storing the labels occurring in the leaves. Also, in order to compare the time complexity of this paper, we add $O(n)$ as the initialization of the algorithm, containing the above storing.

²The time complexity represented in [7] is $O((h + H)\sigma)$ time. In order to compare the time complexity of this paper, we add $O(p + t)$ as the initialization of the algorithm.

denote $v \in V(T)$ by $v \in T$. A *rooted tree* is a tree with one vertex r chosen as its *root*, which we denote by $r(T)$.

For each vertex v in a rooted tree with the root r , let $UP_r(v)$ be the unique path from v to r . The *parent* of $v (\neq r)$, which we denote by $par(v)$, is its adjacent vertex on $UP_r(v)$ and the *ancestors* of $v (\neq r)$ are the vertices on $UP_r(v) \setminus \{v\}$. We denote $u < v$ if v is an ancestor of u , and we denote $u \leq v$ if either $u < v$ or $u = v$. The parent and the ancestors of the root r are undefined. We say that u is a *child* of v if v is the parent of u , and u is a *descendant* of v if v is an ancestor of u . We denote the set of all children of v by $ch(v)$. Two vertices with the same parent are called *siblings*. A *leaf* is a vertex having no children and we denote the set of all the leaves in T by $lv(T)$. We call a vertex that is not a leaf an *internal vertex*.

For a rooted tree $T = (V, E)$ and a vertex $v \in T$, the *complete subtree of T at v* , denoted by $T(v)$, is a rooted tree $S = (V', E')$ such that $r(S) = v$, $V' = \{w \in V \mid w \leq v\}$ and $E' = \{(u, w) \in E \mid u, w \in V'\}$.

The *height* $h(v)$ of a vertex v is defined as $|UP_r(v)| - 1$ and the *height* $h(T)$ of T is the maximum height for every vertex $v \in T$. The *degree* $d(v)$ of a vertex v is the number of the children of v , and the *degree* $d(T)$ of T is the maximum degree for every vertex in T .

We say that a rooted tree is *ordered* if a left-to-right order among siblings is given; *Unordered* otherwise. For a fixed finite alphabet Σ , we say that a tree is *labeled over Σ* if each vertex is assigned a symbol from Σ . We denote the label of a vertex v by $l(v)$, and sometimes identify v with $l(v)$. In this paper, we call a rooted labeled unordered tree over Σ a *tree*, simply.

In this paper, we often represent a rooted labeled unordered tree as a rooted labeled *ordered* tree under a fixed order of siblings. Then, for a rooted labeled ordered tree T , a vertex v in T and its children v_1, \dots, v_i , the *postorder traversal* of $T(v)$ is obtained by first visiting $T(v_k)$ ($1 \leq k \leq i$) and then visiting v . The *postorder number* of $v \in T$ is the number of vertices preceding v in the postorder traversal of T .

Definition 1: Let T and S be trees.

- 1) We say that T is a *subtree* of S , denoted by $T \preceq S$, if T is a tree such that $V(T) \subseteq V(S)$ and $E(T) = \{(v, w) \in E(S) \mid v, w \in V(T)\}$.
- 2) We say that T and S are *isomorphic*, denoted by $T \simeq S$, if $T \preceq S$ and $S \preceq T$.
- 3) We say that T is a *subtree isomorphism* of S , denoted by $T \trianglelefteq S$, if there exists a tree $S' \preceq S$ such that $T \simeq S'$.

In this paper, we deal with a *subtree isomorphism problem* of P for T whether or not $P \trianglelefteq T$ for trees P and T . We call P a *pattern tree* and T a *text tree*. Then, the following theorem holds.

Theorem 1 (Shamir & Tsur [10]): Let P and T be trees where $p = |P|$ and $t = |T|$. Then, the problem of determining whether or not $P \trianglelefteq T$ is solvable in $O(p^{1.5}t/\log p)$ time.

As the restricted form of trees, we introduce a *rooted labeled caterpillar* (a *caterpillar*, for short) as follows.

Definition 2: We say that a tree is a *caterpillar* (cf. [3]) if it is transformed to a rooted path after removing all the leaves in it. For a caterpillar C , we call the remained rooted path a *backbone* of C and denote it by $bb(C)$.

It is obvious that $r(C) = r(bb(C))$ and $V(C) = V(bb(C)) \cup lv(C)$ for a caterpillar C , that is, every vertex in a caterpillar is either a leaf or an element of the backbone.

III. ALGORITHMS FOR SUBCATERPILLAR ISOMORPHISM

In this section, we focus on a *subcaterpillar isomorphism* that is a subtree isomorphism when P is a caterpillar. In other words, we focus on the problem of whether or not $P \trianglelefteq T$ for a caterpillar P and a tree T . Throughout of this section, we refer $p = |P|$, $t = |T|$, $h = h(P)$, $H = h(T)$, $D = d(T)$ and $\sigma = |\Sigma|$.

For a pattern caterpillar P , we refer the backbone of P to a sequence $\langle v_1, \dots, v_n \rangle$ such that $(v_i, v_{i+1}) \in E(P)$ and $v_n = r(P)$. We denote the children of v_i by $ch(v_i)$.

On the other hand, for a text tree T , we refer the vertices in T to w_1, \dots, w_m in postorder traversal. We denote the height of w_j by $h(w_j)$ and the set of children of w_j by $ch(w_j)$.

Let P be a pattern caterpillar and T a text tree such that $P \trianglelefteq T$. Also let $P' \preceq T$ be a subcaterpillar in T such that $P \simeq P'$ and $bb(P') = \langle v'_1, \dots, v'_n \rangle$, where $v'_n = r(P')$. Then, we call the postorder number j such that $v'_1 = w_j$ in T a *matching position* of P in T .

Example 1: Consider a pattern caterpillar P and a text tree T in Figure 1. Here, the number assigned to every vertex in T denotes the postorder number. Also v_i denotes the backbone. Then, $\{6, 8, 16\}$ is the set of all the matching positions of P in T . The corresponding backbones to P in T are $\langle 6, 8, 9 \rangle$, $\langle 8, 9, 18 \rangle$ and $\langle 16, 17, 18 \rangle$.

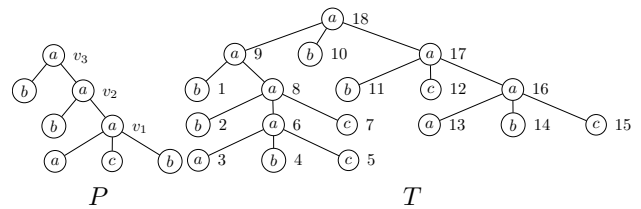


Fig. 1. A pattern caterpillar P and a text tree T in Example 1.

To design the algorithm to determine subcaterpillar isomorphism, we use a *multiset* of labels in order to compare two sets of vertices. A *multiset* on Σ is a mapping $S : \Sigma \rightarrow \mathbb{N}$. For a set V of vertices, we denote the multiset of labels occurring in V by \tilde{V} . Then, it is necessary for the subcaterpillar isomorphism to check whether or not $ch(v_i) \subseteq ch(w_j)$ for $v_i \in bb(P)$ and $w_j \in T$. It is realized to check $\left(\widetilde{ch(v_i)} \right) (a) \leq \left(\widetilde{ch(w_j)} \right) (a)$ for every $a \in \Sigma$ in $O(\sigma)$ time (cf. [9]).

Then, we design the algorithm SUBCATISO in Algorithm 1 to determine whether or not $P \trianglelefteq T$. Here, the algorithm SUBCATISO output all of the matching positions if $P \trianglelefteq T$. Then, it holds that no matching point is output if $P \not\trianglelefteq T$.

```

procedure SUBCATISO( $P, T$ )
  /*  $P$  : caterpillar such that  $bb(P) = \langle v_1, \dots, v_n \rangle$  */
  /*  $T$  : tree consisting of vertices  $w_1, \dots, w_m$  in postorder
  traversal */
1  for  $i = 1$  to  $n - 1$  do  $match[i] \leftarrow \emptyset$ ;
2  for  $j = 2$  to  $m$  do
3    if  $h(w_{j-1}) = h(w_j) + 1$  then
4      /*  $w_j = par(w_{j-1})$  */
5      for  $i = n - 1$  downto  $1$  do
6        if  $match[i] \neq \emptyset$  then
7          foreach  $k \in match[i]$  do
8            if  $h(w_k) = h(w_j) + i$  then
9              /*  $w_j = par(w_k)$  */
10              $match[i] \leftarrow match[i] \setminus \{k\}$ ;
11             if  $l(v_{i+1}) = l(w_j)$  and
12              $ch(v_{i+1}) \subseteq ch(w_j)$  then
13               if  $i + 1 = n$  then output  $k$ ;
14               else
15                  $match[i + 1] \leftarrow$ 
16                  $match[i + 1] \cup \{k\}$ ;
17             if  $l(v_1) = l(w_j)$  and  $ch(v_1) \subseteq ch(w_j)$  then
18               if  $n = 1$  then output  $j$ ;
19               else  $match[1] \leftarrow match[1] \cup \{j\}$ ;

```

Algorithm 1: SUBCATISO.

Example 2: We apply the algorithm SUBCATISO to the pattern caterpillar P and the text tree T in Example 1 in Figure 1. The if-sentence in line 3 works just internal vertices.

- 1) For $j = 6$, since $l(v_1) = a = l(w_6)$ and $ch(v_1) = \{a, b, c\} = ch(w_6)$, 6 is stored to $match[1]$ and then $match[1] = \{6\}$.
- 2) For $j = 8$, since $match[1] \neq \emptyset$, set k to $6 \in match[1]$. Since $h(w_6) = 3 = h(w_8) + 1$, $match[1]$ is changed to \emptyset . Since $l(v_2) = a = l(w_8)$ and $ch(v_2) = \{a, b\} \subseteq ch(w_8)$, 6 is stored to $match[2]$ and then $match[2] = \{6\}$. Also since $ch(v_1) = \{a, b, c\} = ch(w_8)$, 8 is stored to $match[1]$ and then $match[1] = \{8\}$.
- 3) For $j = 9$, since $match[2] \neq \emptyset$, set k to $6 \in match[2]$. Since $h(w_6) = 3 = h(w_9) + 2$, $match[2]$ is changed to \emptyset . Since $l(v_3) = a = l(w_9)$ and $ch(v_3) = \{a, b\} = ch(w_9)$, 6 is output. Also since $match[1] \neq \emptyset$, set k to $8 \in match[1]$. Since $h(w_8) = 2 = h(w_9) + 1$, $match[1]$ is changed to \emptyset . Since $l(v_2) = a = l(w_9)$ and $ch(v_2) = \{a, b\} \subseteq ch(w_8)$, 8 is stored to $match[2]$ and then $match[2] = \{8\}$.
- 4) For $j = 16$, since $ch(v_1) = \{a, b, c\} = ch(w_{16})$, 16 is stored to $match[1]$ and then $match[1] = \{16\}$.
- 5) For $j = 17$, since $match[1] \neq \emptyset$, set k to $16 \in match[1]$. $match[1]$ is changed to \emptyset . Since $l(v_2) = a = l(w_{17})$ and $ch(v_2) = \{a, b\} \subseteq ch(w_{17})$, 16 is stored to $match[2]$ and then $match[2] = \{8, 16\}$.
- 6) For $j = 18$, since $match[2] \neq \emptyset$, set k to 8 and 16. For $k = 8$, since $h(w_8) = 2 = h(w_{18}) + 2$, $match[2]$

is changed to $\{16\}$. Since $l(v_3) = a = l(w_{18})$ and $ch(v_3) = \{a, b\} \subseteq ch(w_{18})$, 8 is output. Also, for $k = 16$, since $h(w_{16}) = 2 = h(w_{18}) + 2$, $match[2]$ is changed to \emptyset . As the same reason, 16 is output.

Hence, the set of all the matching positions of P in T is $\{6, 8, 16\}$. As summarising, Table I illustrates the transition of $match[i]$ for the algorithm SUBCATISO.

TABLE I
THE TRANSITION OF $match[i]$ FOR THE ALGORITHM SUBCATISO.

	$j = 6$	$j = 8$	$j = 9$	$j = 16$	$j = 17$	$j = 18$
$match[1]$	6	8	\emptyset	16	\emptyset	\emptyset
$match[2]$	\emptyset	6	8	8	8, 16	\emptyset
output			6			8, 16

Theorem 2: Let P be a caterpillar and T a tree. Then, the algorithm SUBCATISO correctly outputs all of the matching positions of P in T in $O(p + tDh\sigma)$ time and $O(Dh)$ space.

Proof: First, we show the correctness of the algorithm SUBCATISO. The matching point of P in T is the internal vertices of T . Then, the algorithm SUBCATISO first stores the candidate j of the matching point corresponding to v_1 to $match[1]$ if $l(v_1) = l(w_j)$ and $ch(v_1) \subseteq ch(w_j)$ (line 14). Then, for the current j , the algorithm SUBCATISO removes the candidate k from $match[i]$ if w_j is an ancestor of w_k (line 8) and stores k to $match[i + 1]$ if $l(v_{i+1}) = l(w_j)$, $ch(v_{i+1}) \subseteq ch(w_j)$ and $i < n - 1$ (line 12). If $i = n - 1$, then the algorithm SUBCATISO outputs k (line 10).

Hence, every output k at line 10 satisfies that $l(v_i) = l(par^{i-1}(w_k))$ and $ch(v_i) = ch(par^{i-1}(w_k))$ for every i ($1 \leq i \leq n$), where $par^0(v) = v$ and $par^{i+1}(v) = par(par^i(v))$. As a result, the algorithm SUBCATISO outputs all of the matching points of P in T .

Next, consider the complexity of the algorithm SUBCATISO. As preprocessing, it is necessary to store $ch(v_i)$ for v_i in P and $ch(w_j)$ for internal vertex w_j in T in $O(p)$ time and $O(t)$ time, respectively. Also it is necessary to initialize $match$ in $O(h)$ time. For the for-loop between lines 2 and 12 in the algorithm SUBCATISO, the line 3 works at just internal vertices in T . Since $n = h$ and $|match[i]| \leq D$ ($1 \leq i \leq n - 1$), the foreach-loop between lines 6 and 12 is iterated at most $O(hD)$ times. Since we can check $ch(v_{i+1}) \subseteq ch(w_j)$ in $O(\sigma)$ time, the algorithm SUBCATISO executes the foreach-loop is $O(hD\sigma)$ time. Then, the for-loop is executed in $O(tDh\sigma)$ time. Hence, the total running time of the algorithm SUBCATISO is $O(p + t + h + tDh\sigma) = O(p + tDh\sigma)$ time. The total space is the space spent by the array $match[i]$ for every i ($1 \leq i \leq n - 1$), which is bounded by $O(Dh)$. ■

In order to reduce the searching time in $match[i]$ for every i ($1 \leq i \leq n - 1$) of the algorithm SUBCATISO, we design another algorithm SUBCATISO2 in Algorithm 2.

The main difference between the algorithms SUBCATISO and SUBCATISO2 is that the index i accessed to the array $match$ is determined by $height[h_{j-1}]$ without accessing to $match[i]$ for every i ($1 \leq i \leq n - 1$).

```

procedure SUBCATISO2( $P, T$ )
  /*  $P$  : caterpillar such that  $bb(P) = \langle v_1, \dots, v_n \rangle$  */
  /*  $T$  : tree consisting of vertices  $w_1, \dots, w_m$  in postorder
  traversal */
  1 for  $i = 1$  to  $n$  do  $match[i] \leftarrow \emptyset$ ;
  2 for  $j = 1$  to  $m$  do  $current(j) \leftarrow 0$ ;
  3 for  $h = 1$  to  $h(T) - 1$  do  $height[h] \leftarrow \emptyset$ ;
  4 for  $j = 2$  to  $m$  do
  5    $h_j \leftarrow h(w_j)$ ;  $h_{j-1} \leftarrow h(w_{j-1})$ ;
  6   if  $h_{j-1} = h_j + 1$  then
  7     /*  $w_j = par(w_{j-1})$  */
  8     foreach  $k \in height[h_{j-1}]$  do
  9        $height[h_{j-1}] \leftarrow height[h_{j-1}] \setminus \{k\}$ ;
 10      if  $h(w_k) = h_j + current(k)$  then
 11        /*  $w_j = par(w_k)$  */
 12         $i \leftarrow current(k)$ ;
 13         $match[i] \leftarrow match[i] \setminus \{k\}$ ;
 14         $current(k) \leftarrow 0$ ;
 15        if  $l(v_{i+1}) = l(w_j)$  and
 16         $ch(v_{i+1}) \subseteq ch(w_j)$  then
 17          if  $i + 1 = n$  then output  $k$ ;
 18          else
 19             $match[i + 1] \leftarrow$ 
 20             $match[i + 1] \cup \{k\}$ ;
 21             $height[h_j] \leftarrow height[h_j] \cup \{k\}$ ;
 22             $current(k) \leftarrow i + 1$ ;
 23
 24      if  $l(v_1) = l(w_j)$  and  $\widetilde{ch}(v_1) \subseteq \widetilde{ch}(w_j)$  then
 25        if  $n = 1$  then output  $j$ ;
 26        else
 27           $match[1] \leftarrow match[1] \cup \{j\}$ ;
 28           $height[h_j] \leftarrow height[h_j] \cup \{j\}$ ;
 29           $current(j) \leftarrow 1$ ;

```

Algorithm 2: SUBCATISO2.

Example 3: We apply the algorithm SUBCATISO2 to the pattern caterpillar P and the text tree T in Example 1 in Figure 1. Then, Table II illustrates the transitions of $match[i]$ and $height[j]$ for the algorithm SUBCATISO2.

TABLE II
THE TRANSITIONS OF $match[i]$ AND $height[j]$ FOR THE ALGORITHM SUBCATISO2.

	$j = 6$	$j = 8$	$j = 9$	$j = 16$	$j = 17$	$j = 18$
$match[1]$	6	8	\emptyset	16	\emptyset	\emptyset
$match[2]$	\emptyset	6	8	8	8, 16	\emptyset
<i>output</i>			6			8, 16
$height[1]$	\emptyset	\emptyset	8	8	8, 16	\emptyset
$height[2]$	\emptyset	6, 8	\emptyset	16	\emptyset	\emptyset
$height[3]$	6	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset

- 1) For $j = 6$, by lines 3 and 13, 6 is stored to $match[1]$ and $height[3]$, and $current(6)$ is set to 1.
- 2) For $j = 8$, by lines 6 and 7, 6 is selected as $k \in height[3]$. Since $h(w_6) = 3 = 2 + 1 = h_8 + current(6)$, 6 is deleted from $match[1]$. By line 13, 6 is stored to $match[2]$ and $height[2]$, and $current(6)$ is set to 2. By line 18, 8 is stored to $match[1]$ and $height[2]$, and

$current(8)$ is set to 1.

- 3) For $j = 9$, by lines 6 and 7, 6 and 8 are selected as $k \in height[2]$. For $k = 6$, by line 9, i is set to $2 = current(6)$ and 6 is deleted from $match[2]$. By lines 13 and 14, 6 is output. Also, for $k = 8$, by line 9, i is set to $1 = current(8)$ and 8 is deleted from $match[1]$. By line 13, 8 is stored to $match[2]$ and $height[1]$, and $current(8)$ is set to 2.
- 4) For $j = 16$, by lines 3 and 13, 16 is stored to $height[2]$ and $match[1]$, and $current(16)$ is set to 1.
- 5) For $j = 17$, by lines 6 and 7, 16 is selected as $k \in height[2]$. By line 9, i is set to $1 = current(16)$ and 16 is deleted from $match[1]$. By line 13, 16 is stored to $match[2]$ and $height[1]$, and $current(8)$ is set to 2.
- 6) For $j = 18$, by lines 6 and 7, 8 and 16 are selected as $k \in height[1]$. For $k = 8$, by line 9, i is set to $2 = current(8)$ and 8 is deleted from $match[2]$. By lines 13 and 14, 8 is output. Also, for $k = 16$, by the same reason, 16 is output.

Theorem 3: Let P be a caterpillar and T a tree. Then, the algorithm SUBCATISO2 correctly outputs all of the matching positions of P in T in $O(p + tD\sigma)$ time and $O(D(h + H))$ space.

Proof: The difference between the algorithms SUBCATISO and SUBCATISO2 is the usage of $current$ and $height$ without accessing to $match[i]$ for every i ($1 \leq i \leq n - 1$).

For the selected $k \in height[h_{j-1}]$ at line 7, $current(k)$ is the already processed index i as v_i ($1 \leq i \leq n - 1$). Then, if w_k satisfies the condition at line 13, then $current(k)$ is updated to $i + 1$ at line 17.

On the other hand, for the current j in the for-loop at line 4 and for $k \in height[h_{j-1}]$ at line 7, k is deleted from $height[h_{j-1}]$ at line 8. If $h(w_k) = h_j + current(k)$ at line 9, then h_j is the corresponding height to v_i such that $i = current(k)$. Then, k determines the index i at line 10 to access the array $match[i]$. Furthermore, if v_{i+1} satisfies the condition at line 13, then k is stored to $match[i + 1]$ and $height[h_j]$, and $current(k)$ is updated to $i + 1$.

Hence, the algorithm SUBCATISO2 correctly accesses the array $match[i]$ for every i ($1 \leq i \leq n - 1$). Then, by Theorem 2, the algorithm SUBCATISO2 is correct.

Next, consider the complexity of the algorithm SUBCATISO2. The preprocessing time is $O(p + t)$ from the proof of Theorem 2. It is necessary to initialize $current(j)$ and $height[h]$ in $O(t)$ and $O(H)$, respectively. The foreach-loop between lines 7 and 18 is iterated in $O(D)$ time since $|height[h_{j-1}]| \leq D$, and then the foreach-loop is executed to $O(D\sigma)$ time. Since the for-loop between lines 4 and 22 is executed to $m = t$ time, the total running time of the algorithm SUBCATISO2 is $O(p + t + t + H + tD\sigma) = O(p + tD\sigma)$. The total space of the algorithm SUBCATISO2 is the space spent by the arrays $match[i]$ for every i ($1 \leq i \leq n - 1$) and $height[h]$ for every h ($1 \leq h \leq h(T) - 1$), which is bounded by $O(D(h + H))$. ■

Theorems 2 and 3 imply the following corollary.

Corollary 1: Let P be a caterpillar and T a tree such that $p < t$, $h \ll t$ and $h < H$. Then, the algorithm SUBCATISO determines whether or not $P \trianglelefteq T$ in $O(tD\sigma)$ time and $O(Dh)$ space. Also the algorithm SUBCATISO2 determines whether or not $P \trianglelefteq T$ in $O(tD\sigma)$ time and $O(DH)$ space.

IV. EXPERIMENTAL RESULTS

In this section, we give experimental results of the algorithms SUBCATISO and SUBCATISO2 for both the artificial data and the real data. Here, the computer environment is that OS is Ubuntu 18.04.4, CPU is Intel Xeon E5-1650 v3 (3.50GHz) and RAM is 3.8GB.

A. Artificial data

First, in order to investigate the efficiency of the algorithm SUBCATISO2, we adopt a *binary caterpillar* P_k with height k and the unique label, which is a caterpillar such that every internal vertex has just two children, and a *complete binary tree* T_{2k} with height $2k$ and the unique label, which is a tree such that every internal vertex has just two children and the height of every leaf is just $2k$. It is obvious that $P_k \trianglelefteq T_{2k}$.

Note that the algorithm SUBCATISO2 is more efficient than the algorithm SUBCATISO when the number of the matching points of P in T are large. Then, Table III illustrates the running time of the algorithms SUBCATISO and SUBCATISO2 for P_k and T_{2k} and the number (#match) of matching points of P_k for T_{2k} for $4 \leq k \leq 11$.

TABLE III
THE RUNNING TIME (MSEC.) OF THE ALGORITHMS SUBCATISO AND SUBCATISO2 FOR P_k AND T_{2k} AND THE NUMBER (#MATCH) OF MATCHING POINTS OF P_k FOR T_{2k} FOR $4 \leq k \leq 11$.

P_k	T_{2k}	SUBCATISO	SUBCATISO2	#match
P_4	T_8	4	3	248
P_5	T_{10}	23	21	1,008
P_6	T_{12}	115	98	4,064
P_7	T_{14}	585	473	16,320
P_8	T_{16}	3,256	2,331	65,408
P_9	T_{18}	21,493	12,126	261,888
P_{10}	T_{20}	181,978	67,697	1,048,064
P_{11}	T_{22}	1,579,043	417,140	4,193,280

Table III shows that the algorithm SUBCATISO2 is faster than the algorithm SUBCATISO for P_k and T_{2k} when k is larger.

The number of the matching points of P_{k+1} is about 4 times of those of P_k . On the other hand, the running time of P_8 (resp., P_9 , P_{10} , P_{11}) by the algorithm SUBCATISO is about 5.5 times (resp., about 6.5 times, about 8.5 times, about 8.7 times) of that of P_7 (resp., P_8 , P_9 , P_{10}). Also the running time of P_8 (resp., P_9 , P_{10} , P_{11}) by the algorithm SUBCATISO2 is about 5 times (resp., about 5.2 times, about 5.6 times, about 6.2 times) of that of P_7 (resp., P_8 , P_9 , P_{10}).

B. Real data

Next, we give experimental results for caterpillars and trees in real data. We deal with data for N-glycans and all-

glycans from KEGG³, CSLOGS⁴, dblp⁵ and TPC-H, Auction, Nasa, Protein and University from UW XML Repository⁶. In particular, we deal with the largest 51,546 trees (1%) in dblp (refer to dblp_{1%}). As pattern caterpillars, we deal with non-isomorphic caterpillars in TPC-H, caterpillars obtained by deleting the root in Auction and non-isomorphic caterpillars obtained by deleting the root in Nasa, Protein, and University. Note that we use all the trees as text trees in TPC-H, Auction, Nasa, Protein and University.

Table IV illustrates the information of such caterpillars and trees. Here, #, n , d and h are the number of caterpillars and trees, the average number of vertices, the average degree and the average height.

TABLE IV
THE INFORMATION OF CATERPILLARS AND TREES.

	caterpillars				trees			
	#	n	d	h	#	n	d	h
N-glycans	514	6.40	1.84	4.22	2,124	11.06	2.07	5.38
all-glycans	7,984	4.74	1.49	3.02	10,683	6.38	1.65	3.59
CSLOGS	41,592	5.84	3.05	2.20	59,691	12.93	4.48	3.42
dblp _{1%}	51,395	21.29	20.21	1.04	51,546	21.29	20.18	1.04
SwissProt	6,804	35.10	24.96	2.00	50,000	59.54	31.33	2.76
TCP-H	8	8.63	7.63	1.00	86,805	14.46	13.46	1.00
Auction	259	4.29	3.00	0.71	37	31.00	12.00	3.00
Nasa	33	7.27	5.15	1.64	2,435	195.74	21.53	5.76
Protein	5,150	4.97	3.63	1.16	262,525	81.15	23.27	4.99
University	26	1.35	0.35	0.19	6,739	22.52	11.75	2.31

Then, Table V illustrates the total and average running time (msec.) of the algorithms SUBCATISO and SUBCATISO2 applying to data in Table IV by regarding caterpillars as pattern caterpillars and trees as text trees. Here, #cat denotes the number of pattern caterpillars and #tree denotes the number of text trees. Also the average running time is obtained by dividing the total running time by the total number of pairs, that is, (#cat) × (#tree).

TABLE V
THE TOTAL AND AVERAGE RUNNING TIME (MSEC.) OF THE ALGORITHMS SUBCATISO AND SUBCATISO2 APPLYING TO DATA IN TABLE IV.

	#cat	#tree	SUBCATISO		SUBCATISO2	
			total	ave.	total	ave.
N-glycans	514	2,142	53,969	0.0490	55,638	0.0505
all-glycans	7,894	10,683	1,353,490	0.0159	1,521,891	0.0178
CSLOGS	41,592	59,691	35,681,928	0.0144	42,296,479	0.0170
dblp _{1%}	51,395	51,546	82,881,047	0.0313	85,767,789	0.0324
SwissProt	6,804	50,000	52,694,341	0.1549	53,326,537	0.1568
TCP-H	8	86,805	37,488	0.0540	39,461	0.0568
Auction	259	37	566	0.0591	589	0.0615
Nasa	33	2,435	21,462	0.2671	21,556	0.2683
Protein	5,150	262,525	78,939,972	0.0584	81,660,905	0.0604
University	26	6,739	1,348	0.0077	1,401	0.0080

Furthermore, Table VI illustrates the number (#($P \trianglelefteq T$)) of pairs such that $P \trianglelefteq T$ and its ratio in the total number of

³Kyoto Encyclopedia of Genes and Genomes, <http://www.kegg.jp/>

⁴<http://www.cs.rpi.edu/~zaki/www-new/pmwiki.php/Software/Software>

⁵<http://dblp.uni-trier.de/>

⁶<http://aiweb.cs.washington.edu/research/projects/xmltk/xmldata/www/repository.html>

pairs, and the total and average number (#match) of matching points when $P \trianglelefteq T$.

TABLE VI

THE NUMBER ($\#(P \trianglelefteq T)$) OF PAIRS SUCH THAT $P \trianglelefteq T$ AND ITS RATIO, AND THE TOTAL AND AVERAGE NUMBER (#MATCH) OF MATCHING POINTS WHEN $P \trianglelefteq T$.

	#cat	#tree	$\#(P \trianglelefteq T)$		#match	
			total	ratio (%)	total	ave.
N-glycans	514	2,142	58,128	5.277	96,729	1.664
all-glycans	7,894	10,683	983,163	1.153	1,625,099	1.653
CSLOGS	41,592	59,691	3,201,441	0.129	3,201,441	1.000
dblp _{1%}	51,395	51,546	364,237,724	13.749	364,238,110	1.000
SwissProt	6,804	50,000	20,259,951	5.955	34,465,633	1.701
TCP-H	8	86,805	86,805	12.500	86,805	1.000
Auction	259	37	5,476	57.143	5,476	1.000
Nasa	33	2,435	17,850	22.214	42,687	2.391
Protein	5,150	262,525	2,413,404	0.179	2,515,642	1.042
University	26	6,739	9,260	5.285	9,260	1.000

In contrast to Table III, Table V shows that the algorithm SUBCATISO is faster than the algorithm SUBCATISO2 for real data like as Corollary 1. One of the reasons is that the number of the matching points for real data is much smaller than that for artificial data. In fact, Table VI shows that the average number of matching points for real data when $P \trianglelefteq T$ is less than 2.

Table V also shows that the average running time of both algorithms for Nasa is largest and that for SwissProt is next largest for all the data. The reason is that both the average number of vertices of text trees in Table IV and the average number of matching points in Table VI are larger than other data.

Table VI shows that, for CSLOGS, TCP-H, Auction and University, the number of the matching point is always exactly one when $P \trianglelefteq T$. Then, for the other data as N-glycans, all-glycans, dblp_{1%}, SwissProt, Nasa and Protein, Table VII illustrates the histograms of the number of matching points and the maximum value (max) of matching points when $P \trianglelefteq T$.

TABLE VII

THE HISTOGRAMS FOR THE NUMBER OF MATCHING POINTS AND THE MAXIMUM VALUE (MAX) OF MATCHING POINTS FOR N-GLYCANS, ALL-GLYCANS, DBLP_{1%}, SWISSPROT, NASA AND PROTEIN WHEN $P \trianglelefteq T$.

#match	N-glycans	all-glycans	dblp _{1%}	SwissProt	Nasa	Protein
1	29,909	640,541	364,237,418	13,589,834	10,209	2,344,390
2	20,869	195,221	253	3,828,308	2,692	51,684
3	4,432	72,597	40	1,422,806	3,558	9,862
4	2,804	39,496	6	598,676	590	3,882
5	114	16,172	0	295,015	265	1,394
6	0	9,799	7	161,599	150	1,160
7	0	4,297	0	100,317	99	425
8	0	1,998	0	64,968	59	298
9	0	1,211	0	47,901	48	97
≥ 10	0	1,858	0	150,527	180	212
max	5	21	6	79	1,178	32

Table VII shows that the number of cases whose matching points are more than 1 for SwissProt is largest and that for

all-glycans is next largest for all the data. On the other hand, the maximum value of matching points for Nasa is extremely largest and that for SwissProt is next largest for all the data.

V. CONCLUSION

In this paper, we have investigated the *subcaterpillar isomorphism* and designed two algorithms SUBCATISO running in $O(p + tDh\sigma)$ time and $O(Dh)$ space and SUBCATISO2 running in $O(p + tD\sigma)$ time and $O(D(h + H))$ space, where $p = |P|$, $t = |T|$, $h = h(P)$, $H = h(T)$, $D = d(T)$ and $\sigma = |\Sigma|$. Also we give experimental results for artificial data and real data.

Then, as same as Theorems 2 and 3, we have confirmed that the algorithm SUBCATISO2 is faster than the algorithm SUBCATISO for artificial data whose number of the matching points of P in T are large. On the other hand, we have confirmed that the algorithm SUBCATISO is faster than the algorithm SUBCATISO2 for real data. One of the reason is that the running time of using the array $height[h]$ in the algorithm SUBCATISO2 cannot be absorbed like as Corollary 1 when the number of the matching points is not large.

The reason why we cannot apply the SETH-hardness to subcaterpillar isomorphism is that a caterpillar has a unique backbone. Then, it is a future work to extend a caterpillar to a tree with the bounded number of backbones, in order to avoid to the SETH-hardness of subtree isomorphism [1]. Also it is a future work to extend the algorithms in this paper to *unrooted* subcaterpillar isomorphism like as [10].

REFERENCES

- [1] A. Abboud, A. Backurs, T. D. Hansen, V. v. Williams, O. Zamir: *Subtree isomorphism revisited*, ACM Trans. Algo. **14**, 27 (2018). <https://doi.org/10.1145/3093239>.
- [2] T. Akutsu, D. Fukagawa, M. M. Halldórsson, A. Takasu, K. Tanaka: *Approximation and parameterized algorithms for common subtrees and edit distance between unordered trees*, Theoret. Comput. Sci. **470**, 10–22 (2013). <https://doi.org/10.1016/j.tcs.2012.11.017>.
- [3] J. A. Gallian: *A dynamic survey of graph labeling*, Electorn. J. Combin., DS6 (2018).
- [4] K. Hirata, Y. Yamamoto, T. Kuboyama: *Improved MAX SNP-hard results for finding an edit distance between unordered trees*, Proc. CPM'11, LNCS **6661**, 402–415 (2011). https://doi.org/10.1007/978-3-642-21458-5_34.
- [5] J. E. Hopcroft, R. M. Karp: *An $n^{5/2}$ algorithm for maximum matching in bipartite graphs*, SIAM J. Comput. **2**, 225–231 (1973). <https://doi.org/10.1137/10.1137/0202019>.
- [6] P. Kilpeläinen, H. Mannila: *Ordered and unordered tree inclusion*, SIAM J. Comput. **24**, 340–356 (1995). <https://doi.org/10.1137/S0097539791218202>.
- [7] T. Miyazaki, M. Hagihara, K. Hirata: *Caterpillar inclusion: Inclusion problem for rooted labeled caterpillars*, Proc. ICPRAM'22, 280–287 (2022). <https://doi.org/10.5220/0010826300003122>.
- [8] K. Muraka, T. Yoshino, K. Hirata: *Computing edit distance between rooted labeled caterpillars*, Proc. FedCSIS'18, 245–252 (2018). <http://dx.doi.org/10.15439/2018F179>.
- [9] K. Muraka, T. Yoshino, K. Hirata: *Vertical and horizontal distance to approximate edit distance for rooted labeled caterpillars*, Proc. ICPRAM'19, 590–597 (2019). <https://dx.doi.org/10.5220/0007387205900597>.
- [10] R. Shamir, D. Tsur: *Faster subtree isomorphism*, J. Algo. **33**, 267–280 (1999). <https://doi.org/10.1006/jagm.1999.1044>.
- [11] K. Zhang, T. Jiang: *Some MAX SNP-hard results concerning unordered labeled trees*, Inform. Process. Lett. **49**, 249–254 (1994). [https://doi.org/10.1016/0020-0190\(94\)90062-0](https://doi.org/10.1016/0020-0190(94)90062-0).

Improving N -NEH+ algorithm by using Starting Point method

Radosław Puka

AGH University of Science and Technology in Kraków,
 ul. Gramatyka 10, 30-067 Kraków, Poland
 Email: rpuka@zarz.agh.edu.pl

Bartosz Łamasz, Iwona Skalna

AGH University of Science and Technology
 in Kraków,
 ul. Gramatyka 10, 30-067 Kraków, Poland
 Email: {blamasz, iskalna}@zarz.agh.edu.pl

Abstract—The N -NEH+ algorithm is one of the most efficient construction algorithms for solving the permutation flow-shop problem with the makespan criterion. It extends the well-known NEH heuristic with the N -list technique. In this paper, we propose the Starting Point (SP) method that employs a new strategy for using the N -list technique. The SP method allows to obtain an algorithm that is a combination of NEH and an N -list-based algorithm. Extensive numerical experiments on the standard set of Taillard’s and VRF benchmarks show that the SP method significantly improves the results (average relative percentage deviation) of the NEH and N -NEH+ algorithms.

I. INTRODUCTION

THE permutation flow-shop problem (PFSP) is one of the most famous combinatorial optimization problems in the industry. It can be defined as follows: given two finite sets of m machines $\{M_1, \dots, M_m\}$ and n jobs $\{J_1, \dots, J_n\}$, each of which should go through all the m machines in the same order, the goal is to find the ordering of jobs that minimizes the assumed goal function (makespan, total tardiness, flow time, cost, energy consumption, etc.).

The PFSP with makespan criterion, commonly referred to as $Fm|prmu|C_{\max}$ [1], which is our main concern, is undoubtedly the most frequently investigated scheduling problem. Garey and Johnson [2] have shown that $Fm|prmu|C_{\max}$ is NP-hard if $m \geq 3$. Therefore, various heuristic algorithms have been developed to solve this problem. One of the most popular algorithms for solving $Fm|prmu|C_{\max}$ is the $\mathcal{O}(n^3m)$ NEH construction heuristic proposed by Nawaz, Enscore and Ham [3]. Since 1983 NEH has been commonly regarded as the best heuristic for solving $Fm|prmu|C_{\max}$, and many attempts have been made to improve it. Taillard [4] proposed the so-called *acceleration technique* that reduces the asymptotic time complexity of NEH to $\mathcal{O}(n^2m)$. New criteria were proposed for the initial job classification, e.g., in [5], [6], [7]. The problem of tie-breaking (i.e., how to choose among different subsequences with the same best partial makespan) was considered, e.g., in [8], [9], [10]. One of the most efficient modifications of NEH is the N -NEH+ algorithm recently proposed by Puka et al. [11] which, roughly speaking, combines the NEH heuristic with the N -list technique [11].

This study was conducted under a research project funded by a statutory grant of the AGH University of Science and Technology in Kraków for maintaining research potential.

In this paper, we propose the Starting Point (SP) method that is based on a new strategy of using the N -list technique. The extensive numerical experiments on the standard Taillard’s and VRF benchmarks show that the SP method can significantly improve the results of the N -NEH+ algorithm.

II. STARTING POINT METHOD

The general scheme of the NEH algorithm for solving $Fm|prmu|C_{\max}$ is presented in Algorithm 1.

Algorithm 1 NEH algorithm for solving $Fm|prmu|C_{\max}$

Initial phase: Sort n jobs in non-increasing order of their total processing time and put them into the initial list of jobs $L = \{1, \dots, n\}$.

Insertion phase: Schedule the first job and remove it from L
for $k = 2, \dots, n$ **do**

Insert the job k in the place that minimizes the partial makespan among the k possible ones

Remove job k from the list.

end for

Ruiz and Maroto [12] compared NEH with 25 heuristics for solving PFSPs and it turned out that NEH performed the best both with respect to the quality of the results and the running time. It is no wonder then that a lot of studies on the improvements of NEH performance have been published in the scientific literature.

One of the most efficient NEH-based algorithms for solving $Fm|prmu|C_{\max}$ is the N -NEH+ algorithm [11]. It relies on multiple run of the N -NEH algorithm (see Algorithm 2) that extends NEH with the N -list technique. More specifically, N -NEH+ runs N -NEH for the length of the N -list ranging in the interval $[0, N]$, where $N \leq n - 1$ is the parameter of the N -NEH+ algorithm, and takes the best result of all runs. Let us note that if $N = 1$ (i.e., the N -list technique is not used) the N -NEH+ algorithm is equivalent to NEH.

The Starting Point method proposed in this paper, presented in Algorithm 3, is based on the following strategy: the first N' (N' is a parameter of the SP method) steps are performed as in NEH, and the remaining steps are performed with the use of the N -list technique. For example, given a problem with $n = 100$ and $N' = 0.2n$, the first 20 jobs will be scheduled as in NEH and the remaining 80 jobs as in N -NEH.

Algorithm 2 N -NEH algorithm for solving $Fm|prmu|C_{max}$

Initial phase: Sort n jobs in non-increasing order of their total processing time and put them into the list L .

Insertion phase: Initialize the partial sequence $S = \{1\}$.

Initialize the list L_N of N candidate jobs and remove the respective jobs from L .

for $k = 2, \dots, n$ **do**

Evaluate each job in L_N , put the best job in the respective place in S and remove this job from L_N .

if $L \neq \emptyset$ **then**

Append the first job from L to L_N and remove this job from L .

end if

end for

Algorithm 3 SP method for solving $Fm|prmu|C_{max}$

Initial phase: Sort n jobs in non-increasing order of their total processing time and put them into the list L .

Insertion phase: Initialize the partial sequence $S = \{1\}$.

for $k = 2, \dots, \alpha$ **do**

Insert the job k at the place that minimizes the partial makespan among the k possible ones.

end for

Initialize the list L_N of N candidate jobs and remove the respective jobs from L .

for $k = \alpha + 1, \dots, n$ **do**

Evaluate each job in L_N , put the best job in the respective place in S and remove this job from L_N .

if $L \neq \emptyset$ **then**

Append the first job from L to L_N and remove this job from L .

end if

end for

For the purposes of this work, the Starting Point method will be denoted as $SP(N')N(N)$, where N' is a parameter of the SP method and N is the length of the N -list. Additionally, $SP+(N')N(N)$ will be used to denote the method that relies on running $SP(n')N(N)$ with n' ranging in the interval $[0, N']$; the result of the $SP+(N')N(N)$ method is the best result out of all runs. Similarly, $SP+(N')N+(N)$ will be used to denote the method that runs the $SP(n')N(n')$ method with n' ranging in the interval $[0, N']$ and n'' ranging in the interval $[1, N]$. Let us note that the $SP(0)N$ -NEH+ algorithm (0 means that the SP method is not used) is equivalent to the N -NEH+ algorithm, whereas $SP(N')N+(1)$ is equivalent to NEH.

Since the optimal solution is not known for some instances, the results obtained by a given algorithm are compared with the best solution known so far. The average relative percentage deviation (ARPD) of an algorithm is given as

$$ARPD = \frac{1}{I} \sum_{i=1}^I (C_{max,i} - Best_i) / Best_i, \quad (1)$$

where I is the number of instances, $C_{max,i}$ is the solution

of the algorithm on the instance i , and $Best_i$ is the best solution known so far for this instance. Additionally, for many computed instances, the computational effort of the evaluated algorithm is measured by using the average CPU time (ACPU) given as:

$$ACPU = \left(\sum_{i=1}^I CPU_i \right) / I, \quad (2)$$

where CPU_i is the CPU time of the algorithm on instance i .

The computational experiments that aim to verify the performance of the $SP+(N')N+(N)$ method are presented in the next section. The performance of the method is assessed by using ARPD and ACPU measures.

III. COMPUTATIONAL EXPERIMENT

The performance of the $SP(N')N+(N)$ and $SP+(N')N+(N)$ algorithms (with Taillard's acceleration) was verified on two benchmarks: Taillard's benchmark with 120 instances [13], and VRF benchmark with 240 Small (S) and 240 Large (L) instances [14]. The algorithm was implemented in C# and all the computations were carried out on a computer with two Intel Xeon E5-2660 v4 CPUs (14 cores, each with 2.0 GHz base clock speed).

The results of $SP(N')N+(N)$ (Table I) show that the use of the N -list technique after performing N' steps of insertion phase of NEH can improve the results of the N -NEH+ algorithm, i.e., the $SP(N')N+(N)$ method can outperform N -NEH+. However, if N' is too large ($N' > 0.3n$), the probability of obtaining worse results is greater than in the case of $N' = 0$. The most interesting results are obtained for $N' = 0.1n$ and $N' = 0.2n$. However, it should be noted that improving the results is not possible for all lengths of the N -list.

The results of $SP+(N')N+(N)$ (Table II) show that the greatest decrease in the ARPD values can be observed for smaller values of N' . It can also be observed that for $N' = 0.1n$, the average improvement of N -NEH+ results is 5.9% and for $N' = 0.3n$, the average decrease in the ARPD value is greater than 11.6%. For all the considered benchmarks, a regularity regarding the effectiveness of using the $SP+(N')N+(N)$ method can also be noticed: the longer the N -list, the higher the average percentage decrease in the ARPD value.

From Table III it can be concluded that the larger N' , the shorter computational time of the $SP(N')N+(N)$ algorithm. The computational time also decreases when increasing the length of the N -list. The average decrease in computational time between $SP(0)N+(2)$ (i.e., $N+(2)$) and $SP(0.9n)N+(2)$ ranges from 14% to 16%, and the average decrease in computational time between $SP(0)N+(16)$ (i.e., $N+(16)$) and $SP(0.9n)N+(16)$ ranges from 63% to 69%. Table IV shows that the computational time of $SP+(N')N+(N)$ increases both with N and N' , however the increase with N is much faster.

The ARPD and ACPU values are also shown in Figures 1–3, where the y -axis represents ARPD and x -axis represents ACPU (in logarithmic scale) of the $SP+(N')N+(N)$, NEH

TABLE I
APRD[%] FOR SP(N')N+(N) METHOD ON TAILLARD, VRF S AND VRF L INSTANCES; BEST VALUES FOR EACH $N > 1$ ARE MARKED IN BOLD

Bench.	N	N'									
		0	0.1n	0.2n	0.3n	0.4n	0.5n	0.6n	0.7n	0.8n	0.9n
Tai	1	3.33	3.33	3.33	3.33	3.33	3.33	3.33	3.33	3.33	3.33
	2	2.95	2.96	2.93	2.97	2.96	2.93	2.96	3.07	3.10	3.27
	4	2.55	2.65	2.55	2.59	2.59	2.62	2.69	2.87	2.99	3.22
	8	2.32	2.33	2.29	2.35	2.37	2.41	2.52	2.68	2.91	3.18
	16	2.20	2.16	2.17	2.21	2.24	2.29	2.39	2.60	2.86	3.16
VRF S	1	3.84	3.84	3.84	3.84	3.84	3.84	3.84	3.84	3.84	3.84
	2	3.32	3.32	3.31	3.40	3.47	3.45	3.54	3.63	3.71	3.80
	4	2.95	2.94	2.99	3.03	3.09	3.19	3.34	3.41	3.59	3.78
	8	2.64	2.66	2.67	2.73	2.85	3.01	3.14	3.31	3.53	3.78
	16	2.47	2.42	2.48	2.56	2.70	2.86	3.06	3.27	3.53	3.78
VRF L	1	3.33	3.33	3.33	3.33	3.33	3.33	3.33	3.33	3.33	3.33
	2	3.00	3.00	3.01	3.01	3.03	3.04	3.08	3.11	3.15	3.21
	4	2.67	2.65	2.68	2.72	2.72	2.76	2.84	2.89	2.99	3.11
	8	2.39	2.36	2.40	2.41	2.44	2.52	2.59	2.69	2.83	3.02
	16	2.14	2.11	2.12	2.14	2.20	2.28	2.36	2.50	2.69	2.95

TABLE II
APRD[%] FOR SP+(N')N+(N) METHOD ON TAILLARD, VRF S AND VRF L INSTANCES

Bench.	N	N'								
		0.1n	0.2n	0.3n	0.4n	0.5n	0.6n	0.7n	0.8n	0.9n
Tai	2	2.82	2.74	2.66	2.63	2.58	2.50	2.49	2.48	2.47
	4	2.41	2.31	2.23	2.19	2.15	2.12	2.12	2.11	2.11
	8	2.15	2.04	1.98	1.94	1.93	1.91	1.89	1.89	1.89
	16	2.02	1.90	1.85	1.82	1.81	1.79	1.78	1.78	1.78
	2	3.15	3.01	2.93	2.88	2.84	2.81	2.80	2.78	2.78
VRF S	4	2.74	2.62	2.54	2.50	2.47	2.46	2.44	2.43	2.43
	8	2.45	2.32	2.24	2.20	2.19	2.18	2.17	2.16	2.16
	16	2.24	2.12	2.04	2.02	2.00	2.00	1.99	1.99	1.99
	2	2.89	2.84	2.80	2.77	2.74	2.73	2.73	2.72	2.72
VRF L	4	2.55	2.51	2.49	2.46	2.45	2.45	2.45	2.45	2.45
	8	2.28	2.24	2.22	2.21	2.20	2.20	2.19	2.19	2.19
	16	2.05	2.00	1.98	1.97	1.97	1.97	1.96	1.96	1.96

TABLE III
ACPU[MS] OF SP(N')N+(N) METHOD ON TAILLARD, VRF S AND VRF L INSTANCES

Bench.	N	N'									
		0	0.1n	0.2n	0.3n	0.4n	0.5n	0.6n	0.7n	0.8n	0.9n
Tai	1	38.5	36.3	36.3	36.3	36.2	36.9	36.2	36.2	36.4	36.8
	2	89.9	87.1	86.6	85.7	85.0	84.2	81.8	79.8	78.1	75.7
	4	235.8	231.9	229.2	225.2	218.2	209.5	200.4	187.5	175.1	160.6
	8	702.6	691.8	678.5	656.7	624.9	585.7	541.7	486.8	430.6	358.1
	16	2304.1	2267.4	2205.8	2107.5	1979.7	1810.0	1614.7	1384.1	1129.0	832.8
VRF S	1	1.4	1.3	1.3	1.2	1.2	1.2	1.2	1.2	1.2	1.2
	2	3.0	2.9	2.9	2.8	2.8	2.7	2.7	2.6	2.5	2.4
	4	7.6	7.5	7.3	7.1	6.8	6.6	6.3	5.8	5.4	4.9
	8	21.5	21.0	20.4	19.7	18.7	17.3	15.8	13.8	11.9	9.9
	16	63.4	62.2	59.8	56.4	52.0	46.4	39.9	32.3	25.2	19.7
VRF L	1	625.5	619.0	618.5	617.5	616.4	616.6	616.2	619.2	615.7	616.7
	2	1488.2	1477.7	1468.2	1452.0	1435.0	1413.1	1385.4	1354.5	1314.8	1276.2
	4	3959.7	3922.1	3868.1	3782.5	3678.1	3539.8	3377.0	3186.0	2965.1	2722.4
	8	11838.0	11714.1	11473.0	11072.6	10566.2	9927.0	9149.9	8261.5	7242.4	6107.0
	16	39163.6	38688.9	37624.2	35965.8	33753.0	30997.0	27669.3	23850.2	19489.1	14607.9

TABLE IV
ACPU[MS] OF SP+(N')N+(N) METHOD ON TAILLARD, VRF S AND VRF L INSTANCES

Bench.	N	N'								
		0.1n	0.2n	0.3n	0.4n	0.5n	0.6n	0.7n	0.8n	0.9n
Tai	2	177.1	263.6	349.3	434.3	518.5	600.2	680.0	758.1	833.8
	4	467.8	697.0	922.2	1140.4	1349.9	1550.3	1737.8	1912.9	2073.5
	8	1394.4	2072.9	2729.7	3354.6	3940.3	4482.0	4968.8	5399.4	5757.5
	16	4571.5	6777.3	8884.8	10864.4	12674.4	14289.1	15673.2	16802.1	17634.9
	2	6.0	8.9	11.6	14.4	17.1	19.8	22.4	24.9	27.3
VRF S	4	15.1	22.4	29.5	36.4	43.0	49.2	55.0	60.4	65.3
	8	42.5	62.9	82.6	101.3	118.7	134.5	148.3	160.1	170.1
	16	125.6	185.3	241.7	293.7	340.1	380.0	412.3	437.5	457.2
	2	2965.9	4434.0	5886.1	7321.1	8734.2	10119.6	11474.1	12788.8	14065.1
VRF L	4	7881.8	11749.8	15532.3	19210.4	22750.2	26127.1	29313.2	32278.2	35000.6
	8	23552.1	35025.1	46097.7	56663.9	66590.9	75740.8	84002.3	91244.7	97351.7
	16	77852.5	115476.7	151442.5	185195.5	216192.5	243861.8	267712.0	287201.0	301809.0

and N -NEH+ algorithms. As can be seen from the figures, for all benchmarks, it is difficult to indicate the parameters for which the SP+ method is dominated by other algorithm. In most cases, the usage of $SP+(N')N+(2)$ does not allow obtaining non-dominated results. The results for the remaining parameters form the Pareto front.

Table V presents the comparison of the APRD and ACPU values of the selected algorithms for solving $Fm|prmu|C_{max}$ (cf. [15]). It is not hard to see that these results confirm the good performance of the $SP+(N')N+(N)$ algorithm. This means that the new strategy of delaying the usage of the N -list allows to improve the results of N -NEH+ and makes the $SP+(N')N+(N)$ even more competitive with FRB algorithms [16]. One may argue that in most cases the $SP+(N')N+(N)$ algorithm is more time consuming than FRB algorithms; however, you should keep in mind that $SP+(N')N+(N)$ is (unlike FRB) a construction algorithm, and what is more, it allows you to obtain the entire population of solutions, not just one solution as in the case of FRB.

TABLE V
APRD[%] AND ACPU[MS] VALUES OF SELECTED ALGORITHMS ON
TAILLARD'S BENCHMARK.

Algoitym	APRD	ACPU	Algoitym	APRD	ACPU
RAER	3.89	132.9	FRB4 ₂	2.33	235.2
RAER-di	3.53	277.5	N+(8)	2.32	702.6
NEH	3.33	38.5	SP+(0.2)N+(4)	2.31	697.0
NEMR	3.16	215.1	SP+(0.3)N+(4)	2.23	922.2
KKER	3.15	127.1	N+(16)	2.20	2304.1
NEHKK1-di	3.15	77.6	SP+(0.4)N+(4)	2.19	1140.4
NEH1-di	3.11	77.9	SP+(0.1)N+(8)	2.15	1394.4
NEHKK2	3.09	39.5	FRB4 ₄	2.13	379.5
NEHR	3.05	133.4	SP+(0.2)N+(8)	2.04	2072.9
NEH-di	3.03	76.9	SP+(0.1)N+(16)	2.02	4571.5
CL_WTS	3.02	1789.9	SP+(0.3)N+(8)	1.98	2729.7
NEMR-di	2.97	386.6	FRB4 ₈	1.95	639.3
N+(2)	2.95	89.9	SP+(0.4)N+(8)	1.94	3354.6
NEHFF	2.9	42.2	FRB2	1.93	1335.3
KKER-di	2.86	261.2	FRB4 ₆	1.91	511.2
NEHR-di	2.85	274.5	SP+(0.2)N+(16)	1.90	6777.3
NEHD-di	2.84	344.4	FRB4 ₁₀	1.87	765.0
SP+(0.1)N+(2)	2.82	177.1	SP+(0.3)N+(16)	1.85	8884.8
SP+(0.2)N+(2)	2.74	263.6	SP+(0.4)N+(16)	1.82	10864.4
SP+(0.3)N+(2)	2.66	349.3	FRB4 ₁₂	1.79	879.5
SP+(0.4)N+(2)	2.63	434.3	FRB3	1.61	10522.9
N+(4)	2.55	235.8	FRB5	1.48	30207.4
SP+(0.1)N+(4)	2.41	467.8			

IV. CONCLUSION

This work proposes a Starting Point method that improves N -list technique-based algorithms for solving the permutation flow-shop problem with makespan criterion. The general idea behind the proposed method is to delay the usage of the N -list technique. The extensive numerical experiments on the standard Taillard's and VRF benchmarks show that for both benchmarks the Starting Point method significantly improves the results of the N -NEH and N -NEH+ algorithms. According to the best knowledge of the authors, the analyzed $SP+(N')N+(N)$ algorithm allows obtaining the best results among the results of existing construction algorithms. It is also worth noting that, unlike the more efficient FRB algorithm (which is not construction algorithm), the $SP+(N')N+(N)$

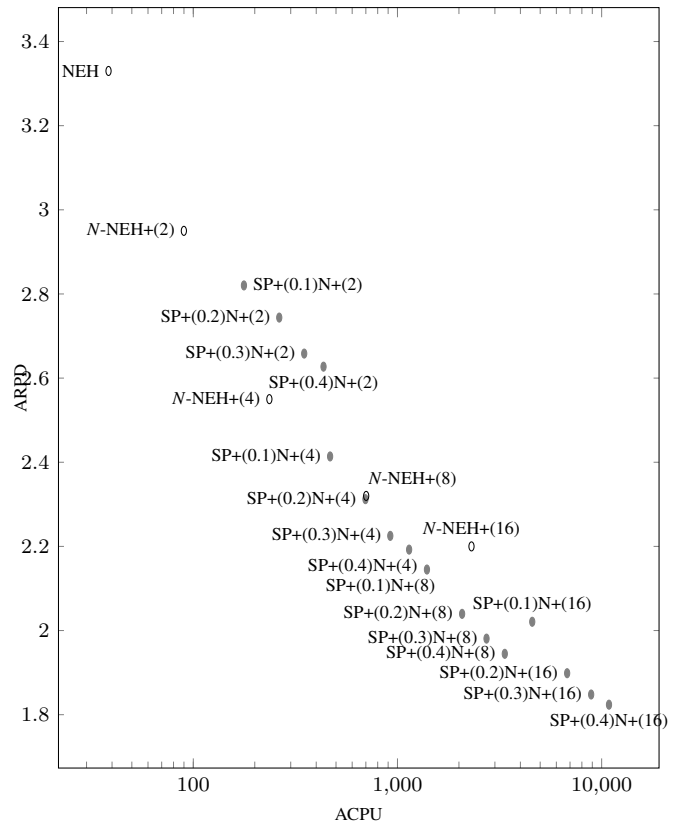


Fig. 1. APRD vs. ACPU (in logarithmic scale) of $SP+(N')N+(N)$ algorithm on Taillard's instances.

algorithm can return not just one but multiple complete rankings. This issue is important because the results of the $SP+(N')N+(N)$ algorithm may be used as the initial population for genetic algorithms. The last important aspect of the $SP+(N')N+(N)$ algorithm is the ease of parallelization of the computations (e.g., for different values of N' and N). For example, the computation time of $SP+(0.3n)N(N)$ run is parallel on 4 cores (one core = one N' value), is equal to the computational time of N -NEH for the same length of the N -list. At the same time, the improvement in the APRD value for different benchmarks ranges from 6.8% to even 17.2%.

REFERENCES

- [1] R. Graham, E. Lawler, J. Lenstra, and A. Kan, "Optimization and approximation in deterministic sequencing and scheduling: a survey," in *Discrete Optimization II*, ser. Annals of Discrete Mathematics, P. Hammer, E. Johnson, and B. Korte, Eds. Elsevier, 1979, vol. 5, pp. 287–326. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S016750600870356X>
- [2] M. Garey and D. Johnson, *Computers and intractability*. San Francisco: W.H. Freeman, 1979, vol. 174.
- [3] M. Nawaz, E. Enscore, and I. Ham, "A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem," *Omega*, vol. 11, no. 1, pp. 91–95, 1983. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/0305048383900889>
- [4] E. Taillard, "Some efficient heuristic methods for the flow shop sequencing problem," *European Journal of Operational Research*, vol. 47, no. 1, pp. 65–74, 1990. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/037722179090090X>

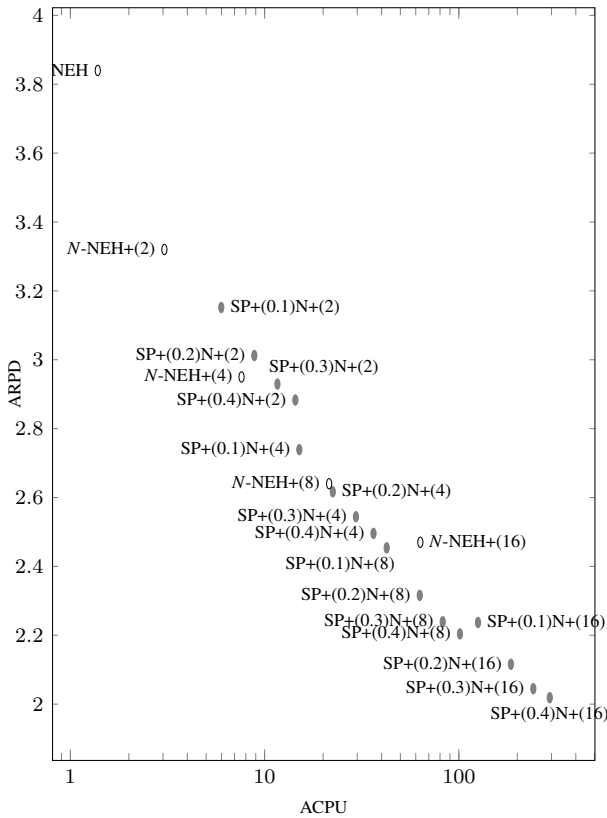


Fig. 2. ARPD vs. ACPU (in logarithmic scale) of $SP+(N')N+(N)$ algorithm on VRF Small instances.

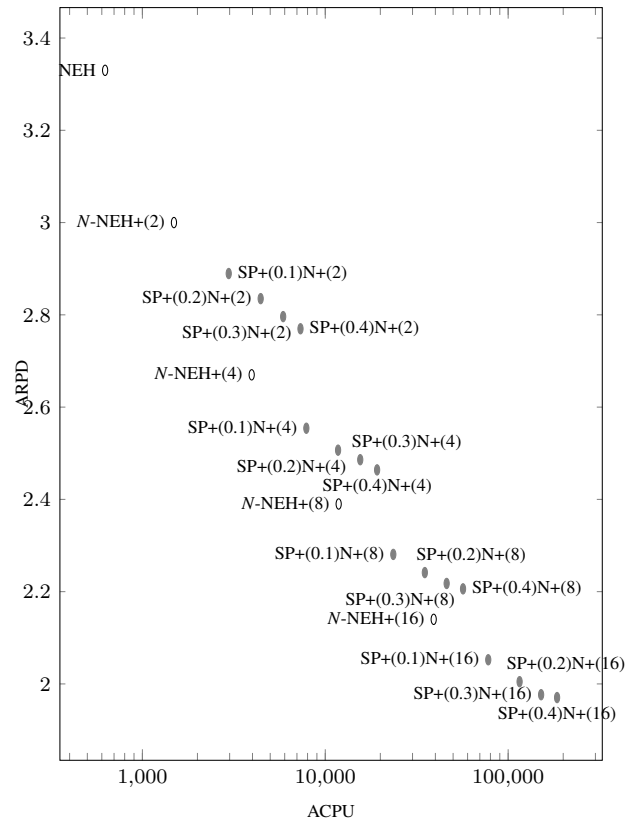


Fig. 3. ARPD vs. ACPU (in logarithmic scale) of $SP+(N')N+(N)$ algorithm on VRF Large instances.

[5] P. Kalczynski and J. Kamburowski, "On the NEH heuristic for minimizing the makespan in permutation flow shops," *Omega*, vol. 35, no. 1, pp. 53–60, 2007.

[6] —, "An improved NEH heuristic to minimize makespan in permutation flow shops," *Computers & Operations Research*, vol. 35, no. 9, pp. 3001–3008, 2008, part Special Issue: Bio-inspired Methods in Combinatorial Optimization.

[7] M. Nagano and J. Moccellini, "A high quality solution constructive heuristic for flow shop sequencing," *Journal of the Operational Research Society*, vol. 53, no. 12, pp. 1374–1379, 2002.

[8] P. Kalczynski and J. Kamburowski, "An empirical analysis of the optimality rate of flow shop heuristics," *European Journal of Operational Research*, vol. 198, no. 1, pp. 93–101, 2009.

[9] V. Fernandez-Viagas and J. Framinan, "On insertion tie-breaking rules in heuristics for the permutation flowshop scheduling problem," *Computers & Operations Research*, vol. 45, pp. 60–67, 2014.

[10] I. Ribas, R. Companys, and X. Tort-Martorell, "Comparing three-step heuristics for the permutation flow shop problem," *Computers & Operations Research*, vol. 37, no. 12, pp. 2062–2070, 2010. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S030505481000050X>

[11] R. Puka, J. Duda, A. Stawowy, and I. Skalna, "N-neh+ algorithm for solving permutation flow shop problems," *Computers & Operations Research*, vol. 132, p. 105296, 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0305054821000885>

[12] R. Ruiz and C. Maroto, "A comprehensive review and evaluation of permutation flowshop heuristics," *European Journal of Operational Research*, vol. 165, no. 2, pp. 479–494, 2005.

[13] E. Taillard, "Benchmarks for basic scheduling problems," *European Journal of Operational Research*, vol. 64, no. 2, pp. 278–285, 1993, project Management and Scheduling. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/037722179390182M>

[14] E. Vallada, R. Ruiz, and J. Framinan, "New hard benchmark for flowshop scheduling problems minimising makespan," *European Journal of Operational Research*, vol. 240, no. 3, pp. 666–677, 2015. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0377221714005992>

[15] V. Fernandez-Viagas, R. Ruiz, and J. Framinan, "A new vision of approximate methods for the permutation flowshop to minimise makespan: State-of-the-art and computational evaluation," *European Journal of Operational Research*, vol. 257, no. 3, pp. 707–721, 2017. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0377221716308074>

[16] S. Rad, R. Ruiz, and N. Boroojerdian, "New high performing heuristics for minimizing makespan in permutation flowshops," *Omega*, vol. 37, no. 2, pp. 331–345, 2009.

Boosting a Genetic Algorithm with Graph Neural Networks for Multi-Hop Influence Maximization in Social Networks

Camilo Chacón Sartori

Artificial Intelligence Research Institute (IIIA-CSIC)
 Campus of the UAB, Bellaterra, Spain
 Email: cchacon@iiia.csic.es

Christian Blum

Artificial Intelligence Research Institute (IIIA-CSIC)
 Campus of the UAB, Bellaterra, Spain
 Email: christian.blum@iiia.csic.es

Abstract—In this paper we solve a variant of the multi-hop influence maximization problem in social networks by means of a hybrid algorithm that combines a biased random key genetic algorithm with a graph neural network. Hereby, the predictions of the graph neural network are used with the biased random key genetic algorithm for a more accurate translation of individuals into valid solutions to the tackled problem. The obtained results show that the hybrid algorithm is able to outperform both the biased random key genetic algorithm and the graph neural network when used as standalone techniques. In other words, we were able to show that an integration of both techniques leads to a better algorithm.

I. INTRODUCTION

Social networks form part of our daily lives. Whether a person is an artist or an engineer, a student or an academic, young or old, or a spartan troll, the person most likely is embedded in one or more social networks. People use social networks to communicate through audiovisual media or text messages, to help others, or even to attack them. In other words, people *influence* other people, either in a positive or in a negative way. Note that the world’s leading technology companies invest huge amounts of money in advertising in social networks. For any social network it is essential to be aware of the transmission of information and its impact.

The identification of a group of users who can influence as many people as possible is a problem called influence maximization (IM). This problem was defined by Kempe et al. in 2003 [1]. By solving the IM problem, a set of adequate people can be identified, for example, for spreading the news about a certain product on a social network. In this way, the dissemination of the product can be supported and eventually maximized. The IM problem has been studied, for example, in the context of dealing with emerging negative opinions [2], social advertising [3], and influence maximization on Twitter for marketing campaigns [4]. Also, different variations of the IM problem have been studied. One example concerns the case of social networks that have a diversity of communities, which implies that there are different types of users who can influence in different ways [5]. Another example is the one of trying to maximize influence in time-evolving social networks [6].

A social network can be viewed as a (directed) graph in which the users are the nodes and user interactions are modeled by arcs. Furthermore, the propagation of influence is often simulated by models such as the independent cascade (IC) model and the linear threshold (LT) model, which might be deterministic or probabilistic. In any case, both models consider one-hop coverage, that is, if a person is covered depends exclusively on its direct neighbors. Although it is common to consider one-hop coverage models in IM problems, the interaction in social networks may also be of multi-hop nature [7].

In this paper we tackle a variant of the IM problem which can be seen as a variant of the classical minimum dominating set problem (MDSP) in a directed graph $G = (V, A)$. In the MDSP, the task is to identify a set of nodes $U \subseteq V$ of minimum cardinality such that for each node $v \in V$ the following holds: either (1) $v \in U$, or (2) it exists at least one node $v' \in U$ such that $(v', v) \in A$, where (v', v) is the directed arc from v' to v . In other words, the classical MDSP considers one-hop coverage. In contrast, in the problem tackled in this paper—known as the multi-hop influence maximization problem (k - d DSP)— d -hop coverage is considered. More specifically, in the k - d DSP the task is to find a set $U \subseteq V$ of cardinality k such that the set $C_U \subseteq V$ of nodes that are covered (or influenced) by U is of maximal cardinality. Hereby, a node v forms part of C_U —that is, v is said to be covered (or influenced) by U —if there exists a node $v' \in U$ such that the shortest directed path from v' to v consists of at most d arcs. As an example, consider Figure 1, where $k = 2$ and the two nodes with a purple color form part of set U , that is, $U = \{v_4, v_5\}$. In case $d = 1$ it holds that $C_U = \{v_4, v_5, v_3, v_6, v_7\}$ because v_3, v_6 and v_7 are in 1-hop distance from a node in U . In other words, the objective function value of U is 5. In case $d = 2$, $C_U = \{v_4, v_5, v_3, v_6, v_7, v_2, v_8, v_{11}\}$ and the objective function value of U would be 8. Finally, in case $d = 3$, $C_U = V$ and the objective function value would be 11.

For a large-scale graph, such as a large social network, exact solutions to the k - d DSP are costly to compute. Therefore, researchers have focused on heuristic and on machine learning

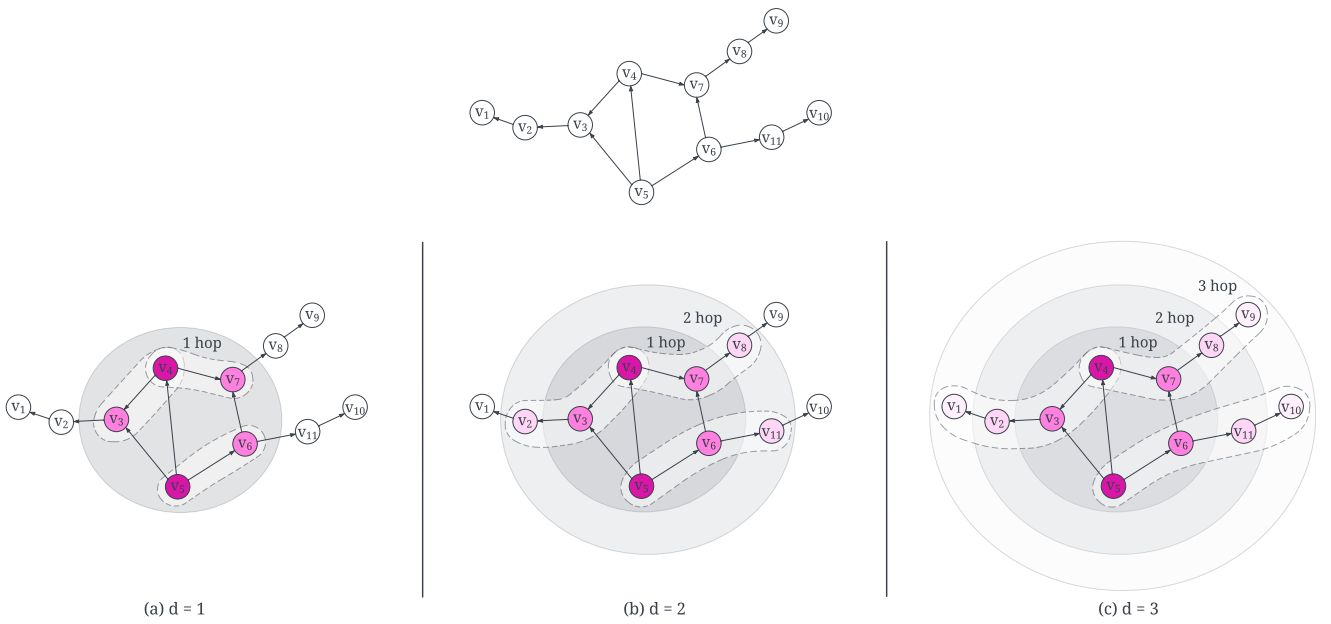


Fig. 1. **Multi-hop influence process.** Given is a directed graph with 11 nodes and 12 arcs (top). Let us assume the k - d DSP is solved with $k = 2$. The two purple nodes (v_4 and v_5) form part of the example solution U . If $d = 1$ (bottom left) then nodes $\{v_3, v_7, v_6\}$ are 1-hop covered by U . If $d = 2$ (bottom center) then nodes $\{v_2, v_3, v_7, v_8, v_6, v_{11}\}$ are 2-hop covered by U . Finally, if $d = 3$ (bottom right), then all remaining nodes of the graph are 3-hop covered by U .

techniques for solving this problem (see Section II). In this paper, we present a novel hybrid algorithm to solve k - d DSP. This algorithm is obtained through an integration of (1) a new graph neural network (GNN) called graph inverse attention network (GRAT) that incorporates the influence of the neighbourhood into the feature embedding of each node [8], and (2) a biased random key genetic algorithm (BRKGA) [9]. More specifically, the information provided by the GNN is used in the BRKGA as greedy information. Our algorithm is evaluated on real-world networks with up to 500.000 nodes and 1 million arcs. Experimental results prove that our hybrid method is at least as good as the two individual techniques in most cases. Therefore, our method is another example for the successful integration of a GNN framework with a metaheuristic to boost the metaheuristics' performance.

The article is organized as follows. Section II introduces prior and related work. In Section III, we provide a more technical description of the k - d DSP. In Section IV, we present our hybrid approach. In Section V, we compare and analyze our hybrid algorithm on real-world data sets. Finally, Section VI concludes the work with some discussions on the utilized type of hybridization.

II. RELATED WORKS

As mentioned before, most of the works on influence maximization (IM) in social networks make use of the independent cascade (IC) and the linear threshold (LT) diffusion models for calculating the influence of solutions. Chen et al. [10], [11] present a fined-tuned heuristic for generating scalable solutions

to the IM problem and an improved runtime in comparison to previous approaches. Jung et al. [12] provide an even faster heuristic for the application to large graphs. Goyal et al. [13] added Monte Carlo simulation in order to obtain an improved heuristic.

Multi-hop influence is a way of measuring the influence of a group of people (set of nodes) in IM problems that has been studied recently. Nguyen et al. [14] proposed a heuristic which takes into account the probability of each node in the network for contributing to a high influence spread. However, the proposed algorithm is only evaluated on small graphs of less than 30.000 nodes. With respect to large graphs, Nguyen et al. [15] presented an alternative heuristic that consists of three phases: pre-optimization to reduce the size of the graph, a construction phase to build a k -dominating set, and post-optimization by removing redundant nodes from the set. This algorithm improves in speed over the one from [16]. However, by doing so it sacrifices performance for a reduction of computation time.

Recently, machine learning techniques have entered the scene to solve combinatorial problems [17]. An early example is S2V-DQN, which is a general reinforcement learning (RL) framework for combinatorial optimization problems in graphs proposed by Khalil et al. [18]. It uses graph neural networks (GNNs) for graph embedding with partial solutions and a *deep Q-network* (DQN) for node selection. This framework is more and more used by the community working with learning techniques for combinatorial optimization.

In the same line, FASTCOVER is a very recent unsupervised

learning framework for solving the k -dDSP by Ni et al. [8]. It uses a multi-layer GNN known as *graph reversed attention network* (GRAT) for generating for each node of a given graph a probability to belong to the optimal solution. The output of FASTCOVER are the k nodes with the highest probability. The authors of [8] show that FASTCOVER outperforms the existing heuristics.

III. PROBLEM DEFINITION

Many optimization problem in social networks can be formalized by modeling the social network as a directed graph $G = (V, A)$, where V is the set of nodes and A is the set of directed arcs present in the graph. This is also the case for the multi-hop influence maximization problem tackled in this paper, denoted by k -dDSP.

The most important concept in this context is the one of the *influence* $I_d(u) \subseteq V$ of a node $u \in V$, which depends on two things:

- 1) Parameter $d \geq 1$, which is part of the problem input.
- 2) A distance measure $dist(u, v)$ between nodes. In the context of this paper, $dist(u, v)$ is defined as the length—in terms of the number or arcs—of the shortest directed path from u to v in G .

With this we can provide the definition of $I_d(u)$ as follows:

$$I_d(u) := \{v \mid dist(u, v) \leq d\} \quad (1)$$

In other words, $I_d(u)$ is the set of all nodes of G that can be reached from u by means of a directed path with at most d arcs. We say that u influences (or covers) all nodes from $I_d(u)$. This definition can naturally be extended to sets of nodes in the following way:

$$I_d(U) := \bigcup_{u \in U} I_d(u) \quad \forall U \subseteq V \quad (2)$$

That is, $I_d(U)$ is the set of all nodes of G that are influenced by at least one node from U .

Valid solutions to the k -dDSP are all sets $U \subseteq V$ such that $|U| \leq k$, that is, any valid solution may consists of at most k nodes. The goal of the k -dDSP is to find a valid solution $U^* \subseteq V$ such that $|I_d(U^*)| \geq |I_d(U)|$ for all valid solutions U to the problem. In other words, the objective function value of a valid solution U is $|I_d(U)|$. In technical terms,

$$\begin{aligned} \max_{U \subseteq V} & |I_d(U)| \\ \text{s.t.} & |U| \leq k \end{aligned} \quad (3)$$

Finally, note that the k -dDSP was proven to be NP-hard in [8], [19].

IV. METHODOLOGY

In this section, we present a novel hybrid algorithm that emerges from the integration between a BRKGA and a GNN framework for solving the k -dDSP in social networks. To begin, we briefly introduce both methods individually. Then we present the developed hybridization strategy.

Algorithm 1 The pseudo-code of BRKGA

Require: a directed graph $G = (V, E)$

Ensure: values for params. $p_{size}, p_e, p_m, prob_{elite}, seed$

```

1:  $P \leftarrow \text{GENERATEINITIALPOPULATION}(p_{size}, seed)$ 
2:  $\text{EVALUATE}(P)$  ▷ dependent part (greedy)
3: while computation time limit not reached do
4:    $P_e \leftarrow \text{ELITESOLUTIONS}(P, p_e)$ 
5:    $P_m \leftarrow \text{MUTANTS}(P, p_m)$ 
6:    $P_c \leftarrow \text{CROSSOVER}(P, p_e, prob_{elite})$ 
7:    $\text{EVALUATE}(P_m \cup P_c)$  ▷ dependent part (greedy)
8:    $P \leftarrow P_e \cup P_m \cup P_c$ 
9: end while
10: return Best solution in  $P$ 

```

A. Biased Random Key Genetic Algorithm

We implemented a Biased Random Key Genetic Algorithm (BRKGA), which is a well-known GA variant for combinatorial optimization. In general, a BRKGA is problem-independent because it works with populations of individuals that are vectors of real numbers (random keys). The problem-dependent part of each BRKGA deals with the way in which individuals are translated into solutions to the tackled problem. The problem-independent pseudo-code of BRKGA is provided in Algorithm 1.

In the following, we first describe the independent or generic part of the algorithm. It starts by invoking function $\text{GenerateInitialPopulation}(p_{size}, seed)$, which generates a population P formed by p_{size} individuals. In case $seed = 0$, all p_{size} individuals are randomly generated. Hereby, each individual $\pi \in P$ is a vector of length $|V|$, where V is the set of nodes from the input graph. For this purpose, the value at position i of π , denoted by $\pi(i)$, is chosen uniformly at random from $[0, 1]$, for all $i = 1, \dots, |V|$. In case $seed = 1$, only $p_{size} - 1$ individuals are randomly generated. The last individual is obtained by defining $\pi(i) := 0.5$ for all $i = 1, \dots, |V|$. Next, the individuals from the initial population are evaluated. This means, each individual $\pi \in P$ is transformed into a valid solution U_π to the k -dDSP, and the value $f(\pi)$ of π is defined as follows: $f(\pi) := |U_\pi|$. The transformation of individuals to valid solutions is discussed below.

Then, at each iteration of the algorithm, the operations to be performed are as follows. First, the best $\max\{\lfloor p_e \cdot p_{size} \rfloor, 1\}$ individuals are copied from P to P_e in function $\text{EliteSolutions}(P, p_e)$. Second, a set of $\max\{\lfloor p_m \cdot p_{size} \rfloor, 1\}$ so-called mutants are generated and stored in P_m . These mutants are random individuals generated in the same way as the random individuals from the initial population. Finally, a set of $p_{size} - |P_e| - |P_m|$ individuals are generated by crossover in function $\text{Crossover}(P, p_e, prob_{elite})$ and stored in P_c .

Each such individual is generated as follows: (1) an elite parent π_1 is chosen uniformly at random from P_e , (2) a second parent π_2 is chosen uniformly at random from $P \setminus P_e$, and (3) an offspring individual π_{off} is generated on the basis of π_1 and

π_2 and stored in P_c . In the context of the crossover operator, value $\pi_{off}(i)$ is set to $\pi_1(i)$ with probability $prob_{elite}$, and to $\pi_2(i)$ otherwise. After generating all new offspring in P_m and P_c , these new individuals are evaluated in function `Evaluate()`; see line 7. Note that the individuals in P_e are already evaluated. Finally, the population of the next generation is determined to be the union of P_e with P_m and P_c .

The evaluation of an individual (see lines 2 and 7 of Algorithm 1) is the problem-dependent part of our BRKGA algorithm. The function that evaluates an individual is often called the *decoder*. In our case, we make use of a simple greedy heuristic which is based on the intuition that nodes with a higher degree (number of neighbors) are more likely to have a high influence than nodes with a lower degree. Hereby, the set of neighbors $N(v_i)$ of a node $v_i \in V$ is defined as follows: $N(v_i) := \{v_j \in V \mid (v_i, v_j) \in A\}$, that is, neighbors of v_i are only those nodes that can be reached via a directed arc from v_i . The greedy value $\phi(v_i)$ of each $v_i \in V$ is defined as follows:

$$\phi(v_i) := |N(v_i)| \cdot \pi(i) \quad (4)$$

In other words, the greedy value of a node v_i is obtained by multiplying the degree of v_i with the numerical value found at position i of the individual to be translated into a solution. Subsequently, solution U_π is obtained by adding the k nodes with the highest greedy values.

Note that, in Section IV-C, greedy function ϕ will be modified in order to obtain a hybrid algorithm.

B. Graph Neural Network Framework

The general objective of a graph neural network (GNN) [20]–[22] is to automatically find patterns in data. In contrast to more classical deep learning techniques, GNNs directly work on graphs. Therefore, they can be used to make predictions about nodes, arcs, or subgraphs without the need for unnecessary transformations of the graph. The crucial idea of GNNs is to iteratively update so-called node representations by combining the representations of a nodes' neighbors with its own representation. Given a graph $G = (V, A)$, $H^l \in \mathbb{R}^{|V| \times C}$ are node attribute matrices, one for each layer $l \in \{0, 1, \dots, L\}$ of the GNN. Note that C is hereby the number of chosen features. Each line in such a matrix is a representation for the respective node. The final goal of a GNN is to learn competent node representations in these matrices.

In order to adapt/train the representations to be useful for a specific task, there are two actions that are successively performed at each GNN layer: (1) *Aggregate*, which aggregates all the information from the neighbors of each node, and (2) *Combine*, which updates the node representations by combining the aggregated information from the neighbors with the current node representation. Based on this, the general framework of a GNN can be specified as follows:

$$\begin{aligned} a_v^l &= \text{AGGREGATE}^l \{ H_u^{l-1} : u \in N(v) \} \\ H_v^l &= \text{COMBINE}^l \{ H_v^{l-1}, a_v^l \} \end{aligned}$$

where $N(v)$ is the set of neighbors of node v . H^K is the node representation matrix for each layer. Once the training process finishes, the final representations can be used for making predictions.

A GNN can be trained, for example, in order to make predictions about the probability of each node to belong to the optimal solution of the k -dDSP. In fact, as mentioned already in the section on related work, such a GNN approach was recently presented in [8]. This GNN—called FASTCOVER (FC)—is an unsupervised GNN framework. FC can be characterized as follows: (1) the features of all nodes are embedded as vector spaces, and the direction of each arc is reversed, (2) a multi-layer GNN known as *graph reversed attention network* (GRAT) assigns each node to its value within $[0, 1]$, and (3) the representations of the GNN are optimized in the training stage through a differentiable loss function over all nodes scores.

The GRAT layer is the heart of FC. In particular, in contrast to a standard *graph attention network* (GAT) [23], the so-called attention mechanism is integrated at the origin nodes instead of the destination nodes. The central idea is that the nodes with more influence are likely to receive a stronger reward. This means a higher probability of getting a potential score.

C. The Hybrid BRKGA Algorithm

Our hybrid algorithm—henceforth called BRKGA+FC—starts with two offline steps. Given a network in which the k -dDSP must be solved, first, all node probabilities are extracted from the trained FC model; note that the prior training process is described in the next section. This probability is denoted by $p_i \in (0, 1]$ for each $v \in V$. Then, the original greedy function $\phi()$ from Eq. 4 is replaced by the following one that incorporates the node probabilities extracted from FC:

$$\phi_{FC}(v_i) := |N(v_i)| \cdot \pi(i) \cdot p_i \quad \forall v_i \in V \quad (5)$$

The hypothesis is that good/correct predictions will bias the algorithm towards the area in the search space in which an optimal solution is located, or, at least, solutions of very high quality. Moreover, we expect the probabilities obtained from FC to undo the bias introduced by the degree of a node, which might sometimes be misleading. The integration process is also shown in Figure 2.

V. EXPERIMENTAL EVALUATION

This section is divided into three parts. First, we will describe the preparation of the data for training and evaluation, and the parameter tuning procedure. Then, the experimental setting and the numerical results of three algorithms will be presented (FC, BRKGA, and BRKGA+FC). In this context note that FC can, of course, be used as a standalone technique by simply adding the k nodes with highest probabilities to the solution. Finally, we will analyse the algorithms graphically by means of so-called search trajectory networks.

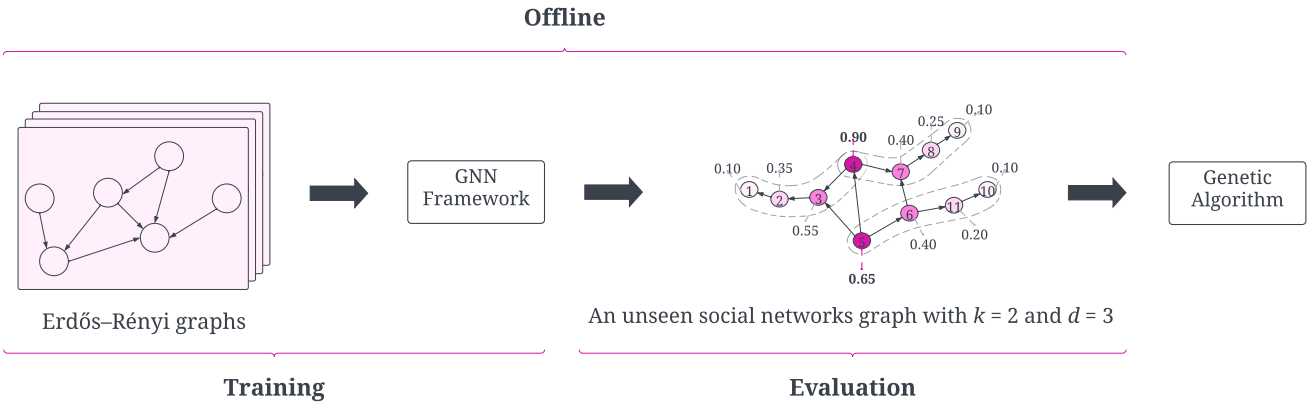


Fig. 2. **Hybridization Process.** The integration of BRKGA with FC starts with two offline steps concerning FC as follows. The training phase begins by using 15 random graphs (Erdős–Rényi). This provides us with a trained version of FC (called GNN Framework in the graphic). Then, the social network in which the k - d DSP is to be solved is presented to FC, which returns probabilities for all nodes of the network to belong to the optimal solution. Finally, the final phase consists of integrating these probabilities into the BRKGA (called Genetic Algorithm in the graphic).

TABLE I
TUNING CONFIGURATION. FINAL PARAMETER SETTING FOR BRKGA AND BRKGA+FC (FOR $k \in \{32, 64, 128\}$)

Parameters	Tuning domain	BRKGA			BRKGA+FC		
		k			k		
		32	64	128	32	64	128
P_{size}	[50, 250]	113	162	132	183	198	137
P_e	[0.1, 2.0]	0.17	0.24	0.25	0.19	0.22	0.2
P_m	[0.3, 5.0]	0.27	0.22	0.14	0.3	0.21	0.21
$prob_{elite}$	[0.01, 0.1]	0.6	0.59	0.58	0.57	0.67	0.67
$seed$	{0, 1}	0	0	0	1	1	1

A. Data Preparation and Tuning Process

We decided to execute experiments for three different values of k , that is, $k \in \{32, 64, 128\}$. For this reason we trained 3 different FC models, one for each value of k . Figure 3 illustrates that each model uses the fixed-parameter $d = 1$. In other words, the same FC model is used for applications of FC and BRKGA+FC for all $d \in \{1, 2, 3\}$. This was done for reducing the computational burden. Nevertheless, in the analysis of the final results we will see that this had some influence on the quality of the node probabilities extracted from the FC models, that is, these probabilities seem to loose accuracy with a growing value of d .

The three FC models (for each value of k) were trained as follows. First, we used 15 Erdős–Rényi graphs [24] with 4000 nodes each, similar to what is presented by the authors of [8]. After the training phase, the probabilities for all 19 social networks used later for the final experimental evaluation are extracted (for each value of $d \in \{1, 2, 3\}$) and stored in text files.

In order to ensure a fair experimental evaluation, both BRKGA and BRKGA+FC were tuned for each value of k

using 10 test graphs. In particular, we used Erdős–Rényi graphs¹ with $n = 25.000$ nodes and an arc probability of $p = 10/n$. The tuning was done using a well-known tool called *irace* [25]. The considered parameter domains together with their finally chosen values are provided in Table I. Note that the number of nodes (25.000) of the test graphs corresponds to approximately the average number of nodes in the networks used for the final experimental evaluation (presented in Section V-B). The size of the tuning graphs is reasonable because the population size parameter in BRKGA is highly dependent on the size of the graphs. In the case of FC, we do not modify the parameters and the configuration as described in [8].

Note that the training phase of FC and the parameter tuning procedure for BRKGA and BRKGA+FC were performed with random graphs to maintain generality.

B. Experimental Evaluation

In this subsection, we apply all three approaches—FC, BRKGA and BRKGA+FC—to 19 real-world social networks from the SNAP library [26]. Each of these networks is a directed, unweighed graph. The sizes of these graphs are provided in Table II (columns $|V|$ and $|A|$).

We use three different values of $k \in \{32, 64, 128\}$. Also for d , the multi-hop influence parameter, we used three different values: $d \in \{1, 2, 3\}$. Note that $k = 64$ was used for the experimental evaluation on social networks of FC [8]. In order to provide a broader experimentation and analysis we also considered two additional values of k : a smaller one (32) and a larger one (128). The reason to not consider values of d greater than 3 is that we were not able to observe substantial differences to the case $d = 3$.

¹Instances can be downloaded from the repository <https://github.com/camilochs/genetic-algorithm-with-gnn>

TABLE II

NUMERICAL RESULTS OBTAINED BY FC, THE BRKGA, AND OUR HYBRID ALGORITHM BRKGA+FC ON 19 WELL-KNOWN SOCIAL NETWORKS. FOR EACH NETWORK THE ALGORITHMS WERE APPLIED FOR $d \in \{1, 2, 3\}$ AND $k \in \{32, 64, 128\}$. FOR $k = 32$ BRKGA+FC WINS IN 73% OF THE CASES; FOR $k = 64$ IN 71%; AND FOR $k = 128$ IN 66%.

Instance	V	E	Distance	$k = 32$			$k = 64$			$k = 128$		
				FC	BRKGA	BRKGA+FC	FC	BRKGA	BRKGA+FC	FC	BRKGA	BRKGA+FC
advogato	6551	51332	1	2338	2464.13	2469.13	2865	2948.67	2948.90	3313	3340.17	3372.33
			2	4069	4139.83	4132.30	4153	4206.83	4207.77	4220	4266.97	4251.13
			3	4268	4279.67	4275.47	4275	4281.80	4280.00	4277	4301.07	4284.00
anybeat	12645	67053	1	8556	8566.80	8570.70	9045	8981.40	9002.83	9650	9537.60	9626.47
			2	11104	11177.10	11205.63	11209	11300.53	11305.83	11384	11371.47	11400.77
			3	11507	11527.17	11526.00	11515	11531.00	11530.33	11556	11542.27	11546.87
brightkite	56739	212945	1	1266	1714.33	1808.67	1954	2483.03	2640.27	3023	3448.00	3711.63
			2	4018	4160.63	4671.50	5444	5088.90	5910.40	6795	6075.13	6891.07
			3	6094	5699.57	6535.13	7530	6349.90	7614.10	8650	7178.47	8189.63
delicious	536108	1365961	1	8522	10860.00	10864.83	12431	15793.53	15792.90	19483	21044.43	21170.37
			2	21119	22341.80	22481.57	26018	26909.07	26995.33	32248	32811.27	33414.13
			3	32000	33041.13	33175.07	36112	36039.43	36328.03	40309	40418.77	41722.63
douban	154908	327162	1	1093	1503.83	1482.30	2117	2649.93	2637.90	3950	4557.10	4565.73
			2	4147	6809.23	6743.50	6801	9516.80	9594.53	10583	12950.27	13093.53
			3	11686	13988.10	14448.00	15938	17548.57	17866.60	20720	21277.43	22368.23
epinions	26588	100120	1	1532	1753.27	1774.70	2198	2333.10	2413.17	3019	3000.47	3170.93
			2	3645	3711.43	3853.17	4271	4086.47	4416.07	4904	4549.13	4897.77
			3	4500	4487.73	4634.20	4948	4661.37	5002.83	5430	4973.10	5334.60
gowalla	196591	950327	1	1998	3296.13	3384.73	3415	4869.60	5122.30	5553	6976.07	7403.00
			2	7509	9723.47	10754.63	10734	12534.07	13628.10	15386	14888.50	15121.67
			3	14247	14913.00	16657.80	18418	17386.73	18966.73	23692	19064.10	22800.10
gplus	23628	39242	1	17498	18077.00	17896.00	22138	22496.93	22167.90	23543	23628.00	23567.00
			2	21277	23077.20	22726.63	23200	23562.93	23172.73	23628	23628.00	23628.00
			3	21636	23271.37	22884.60	23271	23559.80	23169.00	23628	23628.00	23628.00
loc-brightkite	58228	214078	1	8778	9041.33	9047.27	11232	11719.57	11722.57	14749	15058.97	15128.27
			2	37295	38212.40	38267.47	40161	41258.13	41190.97	42929	43777.03	43827.50
			3	52335	52645.00	52744.00	53272	53600.17	53469.27	53783	54123.80	54134.60
sign-Slashdot081106	77350	516575	1	7162	8087.00	8087.00	11362	11999.33	12003.93	17046	17154.80	17524.33
			2	37521	42221.13	42352.03	44291	47782.33	47827.97	47747	51839.47	51796.63
			3	59456	60393.67	60367.30	60709	61148.47	61148.07	61333	61683.10	61701.40
sign-Slashdot090216	81867	545671	1	7232	8127.87	8128.00	11385	12094.27	12108.50	16949	17592.80	17613.43
			2	39841	43723.57	43781.93	46021	49774.13	49832.23	49661	54244.47	54141.53
			3	62964	63840.83	63817.00	64209	64710.97	64723.77	64912	65375.13	65399.37
sign-Slashdot090221	82140	549202	1	7182	8129.00	8129.00	11421	12129.03	12126.33	17010	17642.67	17641.80
			2	39220	43869.37	43982.53	46410	49972.23	49968.17	49917	54408.47	54334.93
			3	62958	64062.17	64036.97	64473	64935.37	64953.90	65145	65589.77	65598.23
sign-bitcoinotc	5881	35592	1	3455	3479.00	3479.00	4010	4038.17	4040.97	4615	4595.70	4617.97
			2	5568	5631.97	5632.60	5645	5715.37	5715.20	5761	5716.83	5781.17
			3	5814	5838.00	5838.03	5834	5839.00	5839.10	5844	5842.00	5844.00
sign-epinions	131828	841372	1	17765	18690.03	18693.50	22933	23569.77	23609.43	28969	29052.87	29284.87
			2	56849	59372.80	59411.70	60208	62238.23	62288.90	63070	64153.33	64309.37
			3	70410	70739.73	70721.93	70829	71021.30	71024.17	71188	71245.93	71309.97
slashdot	70068	358647	1	3419	3722.70	3791.37	4683	5020.93	5083.63	6304	6515.80	6683.03
			2	7849	8302.53	8958.23	9534	9605.07	9771.83	11083	10802.73	11486.17
			3	10537	10527.87	11223.53	11699	11166.57	11810.70	12648	11964.73	12660.03
slashdot-zoo	79120	515581	1	7229	8100.00	8100.00	11460	12021.30	12051.57	17068	17514.53	17515.70
			2	37664	41741.40	41848.30	43850	47347.50	47346.50	47629	51361.57	51334.70
			3	58974	59818.57	59805.10	60176	60600.87	60604.67	60797	61135.87	61140.97
themarket	69413	1644849	1	32088	32320.00	32320.00	35567	35891.10	35908.27	39245	39137.20	39258.97
			2	51299	52117.30	52127.60	52100	52654.87	52665.13	52785	53016.10	53093.40
			3	53506	53598.40	53593.03	53575	53665.10	53666.67	53729	53754.67	53758.90
twitter-follows	404719	713319	1	14063	14548.57	14549.53	26981	27755.17	27760.93	48870	51240.20	50531.97
			2	52319	88442.93	88620.80	83380	117605.90	117692.57	85371	115365.57	135258.13
			3	171902	207181.27	204973.27	203499	213702.93	214154.77	212524	219099.80	221819.07
wiki-elec	7118	107071	1	2146	2167.00	2176.70	2227	2265.63	2268.63	2306	2367.70	2366.47
			2	2328	2354.73	2355.10	2341	2390.00	2388.03	2366	2454.50	2427.53
			3	2331	2357.10	2357.27	2344	2389.53	2389.67	2366	2452.23	2426.47

As FC is a deterministic approach (at least for what concerns the use of the model after training), it was applied exactly once to each of the 19 networks, for each combination of k and d . On the contrary, both BRKGA and BRKGA+FC were applied 30 times to each network and combination of k and d . As a computation time limit of BRKGA and BRKGA+FC we used 900 CPU seconds for each run. All experiments were performed on machines with an Intel(R) Xeon(R) Silver 4210 CPU @ 2.20GHz. The FC framework uses Python 3 and our implementations of BRKGA and BRKGA+FC were coded in C++.

In Table II we compare the results of FC with the average results of BRKGA and BRKGA+FC over 30 runs. The following observations can be made:

- Generally, both BRKGA and BRKGA+FC outperform FC which, in turn, was shown in [8] to outperform the existing heuristics for solving the k -dDSP. The only exceptions are 1 case with $k = 64$ and 9 cases with $k = 128$. This suggests that the performance of FC (when compared to the BRKGA versions) improves with an increasing value of k .
- Even though BRKGA generally outperforms FC, the

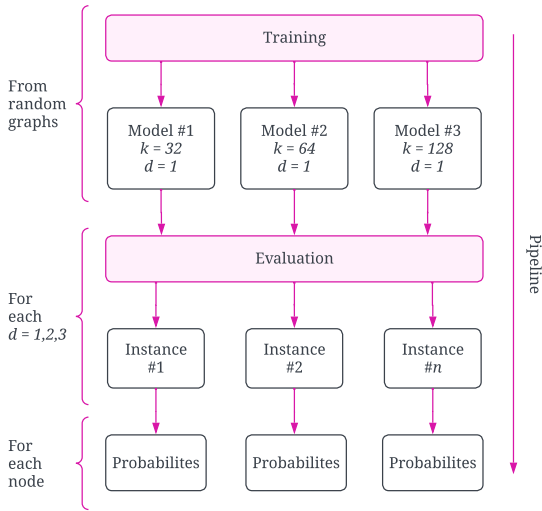


Fig. 3. **Data preparation and pipeline.** The pipeline starts by training three FC models, one for each $k \in \{32, 64, 128\}$. Random graphs (Erdős–Rényi) were used for this purpose. Next, the evaluation of the FC models is performed for each of the 19 instances (social networks), for each value of parameter $d \in \{1, 2, 3\}$. Finally, the obtained probabilities (FC output) are exported and stored in text files.

hybrid algorithm BRKGA+FC generally benefits from the use of the probability information extracted from FC for the translation of individuals into solutions. This advantage of BRKGA+FC over BRKGA is greatest for the smallest value of d (that is, $d = 1$). In this case BRKGA+FC outperforms BRKGA in 73% of all cases.

- The worst performance of BRKGA+FC is obtained for $k = 32$ and $d = 3$ (47% of superiority). This may be due to two possible reasons: (1) FC might find it difficult to detect pattern for rather small values of k ; (2) all our FC models were trained for $d = 1$, which might suggest that our results could be improved by specifically training FC for each value of d .

Summarizing, we can say that making use of information from the GNN framework FC within our BRKGA clearly improves the algorithm.

C. Analysis

There are cases when our hybrid algorithm does not perform as expected. This is the case when the results are similar to the ones of BRKGA, or when they are even worse than the ones of BRKGA. In an attempt to analyse such cases we used the Search Trajectory Networks (STNs) tool from [27], which allows to visualize the trajectories of algorithms in the search space. Moreover, it lets us compare the behavior of more than one metaheuristic. For this analysis we chose three network corresponding to three different cases as outlined in Figure 4. The obtained graphics allow to make the following observations.

- 1) Figure 4 (a). This is a case in which the hybrid algorithm BRKGA+FC does not perform well in comparison to BRKGA. We can see in the graphic that both algorithms are clearly focused on different areas of the search space. In particular, BRKGA is attracted by a certain area of the search space. Nevertheless, the best solution found (red dot), even though it belongs to this part of the search space, it is not close to the area of attraction (see the two larger grey triangles). One hypothesis is that the probabilities provided by the graph neural network framework (FC) for this instance are rather misleading.
- 2) Figure 4 (b). In this case, the performance of both algorithms is comparable. Again, the two algorithm version are focused on different areas of the search space. This time there is a minimal overlap between two of the algorithm trajectories (see the light gray dot in the middle of the graphic). Interestingly, even though both algorithms find a best solution of the same quality, these two solutions are clearly different to each other (see the two red dots).

However, as mentioned before, in a majority of cases BRKGA+FC outperforms BRKGA. Such a case is visualized in the graphic of Figure 4 (c). It can be observed that the trajectory of BRKGA+FC is more bounded and, therefore, it is not dispersed in the search space as it occurs for BRKGA. Moreover, the best solution is found in the area of the search space that attracts BRKGA+FC. This means that, in this case, the information provided by the FC is very useful.

VI. CONCLUSION AND FUTURE WORK

In this work we have devised a hybrid algorithm combining a biased random key genetic algorithm with a graph neural network called FASTCOVER. This was done in the context of an NP-hard combinatorial optimization problem dealing with the maximization of influence spreading in social networks. In particular, our hybrid algorithm makes use of the recommendations provided by FASTCOVER (in the form of probabilities) for translating individuals to valid solutions to the tackled problem. The results have shown that, in a majority of the cases, our hybrid algorithm outperforms both its individual algorithmic components: the biased random key genetic algorithm and FASTCOVER. The experimental evaluation of our approaches was done in the context of 19 real-world social networks.

One opportunity to advance this type of hybridization is to address other problems using a similar integration methodology, especially taking the recent progress of graph representation learning into account.

ACKNOWLEDGMENT

This paper was supported by grant PID2019-104156GB-I00 funded by MCIN/AEI/10.13039/501100011033.

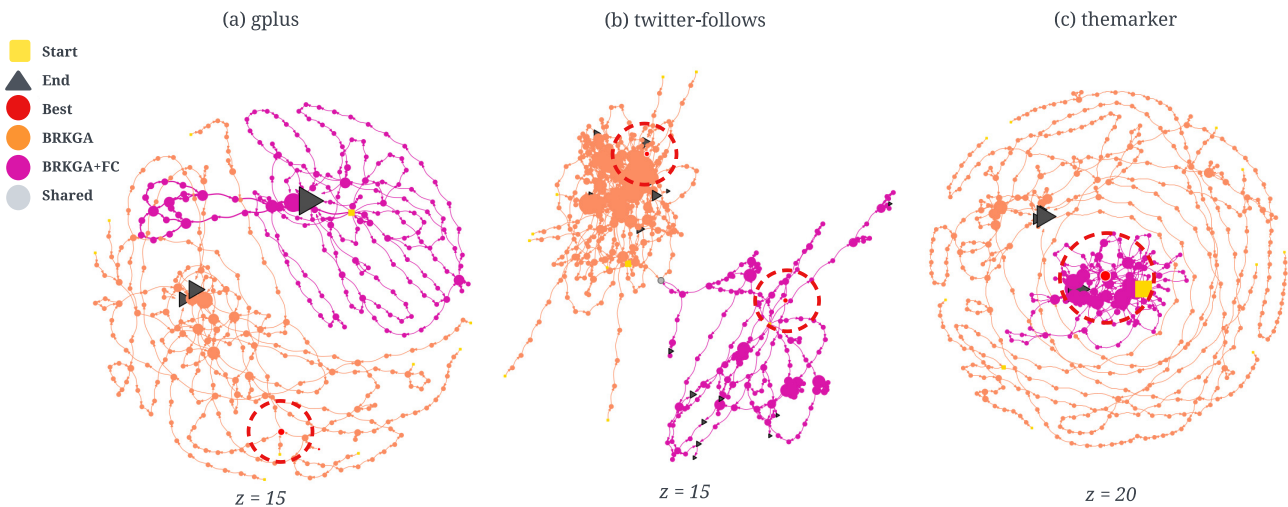


Fig. 4. Search trajectory analysis concerning BRKGA and BRKGA+FC. The three graphics show 10 executions (trajectories) of BRKGA (orange) and BRKGA+FC (pink) for three instances (gplus, twitter-follows, and themarker). The value of z indicates the degree of search space partitioning used to generate the graphics (see [27]). Yellow squares indicate the start of trajectories, while gray triangles indicate their ends. Also, light gray circles indicate that both algorithms passed through this location of the search space, while red circles indicate the best solutions found. (a) A case in which BRKGA is able to outperform BRKGA+FC (gplus). (b) A case in which BRKGA and BRKGA+FC achieve similar results (twitter-follows). (c) A case in which BRKGA+FC outperforms BRKGA (themarker). For each graphic we used the force-directed layout based on physical analogies, not relying on any assumptions about the structure of the networks.

REFERENCES

- [1] D. Kempe, J. Kleinberg, and E. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '03. New York, NY, USA: Association for Computing Machinery, 2003, p. 137–146. [Online]. Available: <https://doi.org/10.1145/956750.956769>
- [2] W. Chen, A. Collins, R. Cummings, T. Ke, Z. Liu, D. Rincon, X. Sun, Y. Wang, W. Wei, and Y. Yuan, *Influence Maximization in Social Networks When Negative Opinions May Emerge and Propagate*, 2011, pp. 379–390. [Online]. Available: <https://epubs.siam.org/doi/abs/10.1137/1.9781611972818.33>
- [3] S. Tang, "When social advertising meets viral marketing: Sequencing social advertisements for influence maximization," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, ser. AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018. [Online]. Available: <https://ojs.aaai.org/index.php/AAAI/article/view/11306>
- [4] Y. Mei, W. Zhao, and J. Yang, "Influence maximization on twitter: A mechanism for effective marketing campaign," in *2017 IEEE International Conference on Communications (ICC)*, 2017, pp. 1–6. [Online]. Available: <https://ieeexplore.ieee.org/document/7996805/>
- [5] J. Li, T. Cai, K. Deng, X. Wang, T. Sellis, and F. Xia, "Community-diversified influence maximization in social networks," *Information Systems*, vol. 92, p. 101522, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0306437920300326>
- [6] H. Zhuang, Y. Sun, J. Tang, J. Zhang, and X. Sun, "Influence maximization in dynamic social networks," in *2013 IEEE 13th International Conference on Data Mining*, 2013, pp. 1313–1318. [Online]. Available: <https://ieeexplore.ieee.org/document/6729640>
- [7] A. Goyal, W. Lu, and L. V. Lakshmanan, "Simpath: An efficient algorithm for influence maximization under the linear threshold model," in *2011 IEEE 11th International Conference on Data Mining*, 2011, pp. 211–220. [Online]. Available: <https://ieeexplore.ieee.org/document/6137225>
- [8] R. Ni, X. Li, F. Li, X. Gao, and G. Chen, "Fastcover: An unsupervised learning framework for multi-hop influence maximization in social networks," 2021. [Online]. Available: <https://arxiv.org/abs/2111.00463>
- [9] J. F. Gonçalves and M. G. C. Resende, "Biased random-key genetic algorithms for combinatorial optimization," *Journal of Heuristics*, vol. 17, no. 5, pp. 487–525, Oct 2011. [Online]. Available: <https://doi.org/10.1007/s10732-010-9143-1>
- [10] W. Chen, Y. Wang, and S. Yang, "Efficient influence maximization in social networks," in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '09. New York, NY, USA: Association for Computing Machinery, 2009, p. 199–208. [Online]. Available: <https://doi.org/10.1145/1557019.1557047>
- [11] W. Chen, Y. Yuan, and L. Zhang, "Scalable influence maximization in social networks under the linear threshold model," in *2010 IEEE International Conference on Data Mining*, 2010, pp. 88–97. [Online]. Available: <https://ieeexplore.ieee.org/document/5693962>
- [12] K. Jung, W. Heo, and W. Chen, "Irie: Scalable and robust influence maximization in social networks," in *2012 IEEE 12th International Conference on Data Mining*, 2012, pp. 918–923. [Online]. Available: <https://ieeexplore.ieee.org/document/6413832>
- [13] A. Goyal, W. Lu, and L. V. Lakshmanan, "Celf++: Optimizing the greedy algorithm for influence maximization in social networks," in *Proceedings of the 20th International Conference Companion on World Wide Web*, ser. WWW '11. New York, NY, USA: Association for Computing Machinery, 2011, p. 47–48. [Online]. Available: <https://doi.org/10.1145/1963192.1963217>
- [14] D.-L. Nguyen, T.-H. Nguyen, T.-H. Do, and M. Yoo, "Probability-based multi-hop diffusion method for influence maximization in social networks," *Wireless Personal Communications*, vol. 93, no. 4, pp. 903–916, Apr 2017. [Online]. Available: <https://doi.org/10.1007/s11277-016-3939-8>
- [15] M. H. Nguyen, M. H. Hà, D. N. Nguyen, and T. T. Tran, "Solving the k-dominating set problem on very large-scale networks," *Computational Social Networks*, vol. 7, no. 1, p. 4, Jul 2020. [Online]. Available: <https://doi.org/10.1186/s40649-020-00078-5>
- [16] Y. Wang, S. Cai, J. Chen, and M. Yin, "A fast local search algorithm for minimum weight dominating set problem on massive graphs," in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, ser. IJCAI'18. AAAI Press, 2018, p. 1514–1522. [Online]. Available: <https://www.ijcai.org/proceedings/2018/210>

- [17] Y. Bengio, A. Lodi, and A. Prouvost, "Machine learning for combinatorial optimization: a methodological tour d'horizon," 2018. [Online]. Available: <https://arxiv.org/abs/1811.06128>
- [18] H. Dai, E. B. Khalil, Y. Zhang, B. Dilkina, and L. Song, "Learning combinatorial optimization algorithms over graphs," 2017. [Online]. Available: <https://arxiv.org/abs/1704.01665>
- [19] P. Basuchowdhuri and S. Majumder, "Finding influential nodes in social networks using minimum k-hop dominating set," in *Applied Algorithms*, P. Gupta and C. Zaroliagis, Eds. Cham: Springer International Publishing, 2014, pp. 137–151. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-04126-1_12
- [20] L. Wu, P. Cui, J. Pei, and L. Zhao, Eds., *Graph Neural Networks: Foundations, Frontiers, and Applications*. Springer, Singapore, 2022. [Online]. Available: <https://link.springer.com/book/10.1007/978-981-16-6054-2>
- [21] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive survey on graph neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 1, pp. 4–24, jan 2021. [Online]. Available: <https://doi.org/10.1109%2Ftnnls.2020.2978386>
- [22] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?" 2018. [Online]. Available: <https://arxiv.org/abs/1810.00826>
- [23] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," 2017. [Online]. Available: <https://arxiv.org/abs/1710.10903>
- [24] P. Erdos and A. Renyi, "On the evolution of random graphs," *Publ. Math. Inst. Hungary. Acad. Sci.*, vol. 5, pp. 17–61, 1960. [Online]. Available: <http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.153.5943>
- [25] M. López-Ibáñez, J. Dubois-Lacoste, L. Pérez Cáceres, M. Birattari, and T. Stützle, "The irace package: Iterated racing for automatic algorithm configuration," *Operations Research Perspectives*, vol. 3, pp. 43–58, 2016. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2214716015300270>
- [26] J. Leskovec and R. Susic, "Snap: A general purpose network analysis and graph mining library," 2016. [Online]. Available: <https://arxiv.org/abs/1606.07550>
- [27] G. Ochoa, K. M. Malan, and C. Blum, "Search trajectory networks: A tool for analysing and visualising the behaviour of metaheuristics," *Applied Soft Computing*, vol. 109, p. 107492, 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1568494621004154>

Stackelberg Strategies for Weighted Load Balancing Games

Neta Stein and Tami Tamir

School of Computer Science

Reichman University (IDC)

Herzliya, Israel

Emails: neta.stein@post.idc.ac.il , tami@idc.ac.il

Abstract—An instance of a weighted Stackelberg load balancing game is given by a set of identical machines, a set of variable-length jobs and a parameter $0 \leq \alpha \leq 1$. A centralized authority, denoted *the leader*, selects a subset of the jobs whose total length is at most an α -fraction of the total length and determines their assignment on the machines. After the controlled jobs are assigned, the remaining jobs join the schedule. They act selfishly, each determining its own assignment.

Our work combines theoretical and experimental results for this setting. We suggest various heuristics for the leader and analyze their performance.

I. INTRODUCTION

IN RESOURCE allocation applications, a set of resources is used by a set of clients (users). For example, in job-scheduling applications, servers process jobs; in communication or transportation networks, traffic is routed on network links. In some settings, the users are assigned to the different resources by a centralized authority. The centralized authority is aware of all clients' requests and can utilize the system in the best possible way. Other resource allocation applications lack a central authority and are managed by multiple strategic users, whose individual payoff is affected by the assignment of other users. As a result, game theory has become an essential tool in the analysis of resource-allocation services. In the corresponding game, every client is a selfish player who aims at maximizing his own utilization.

In this work, we consider systems in which the two models are combined. That is, some of the users obey a central authority and some are selfish. The goal of the system is to assign the centrally controlled users such that the total system performance is optimized. We formulate this goal as an optimization problem via *Stackelberg games*, in which one player, corresponding to the centralized authority, acts as a *leader* and the rest of the players, corresponding to the selfish users, as *followers*. The problem is then to compute a strategy for the leader (a Stackelberg strategy) that cause the followers to react in a way that is as good as possible for the social utility.

We focus on load balancing games in job scheduling systems. The users are jobs having variable weights and the resources are machines. Every job should be assigned on a single machine. The cost of a job depends on the total load on its machine and the social cost is given by the load on the most loaded machine. In centralized systems, this load

balancing scenario is the well-studied *minimum Makespan* problem. It was well-studied also as a game where all the jobs are selfish, however, it has not been studied in an environment with a mixed type of jobs. All previous works on Stackelberg strategies consider non-atomic (splittable) players – every player has a neglected load and the total load can be distributed among the machines in an arbitrary way. Thus, our work initiate the study of Stackelberg strategies for weighted singleton congestion games.

Our goal is to develop Stackelberg strategies that guarantee load balancing and a low social cost. We will consider several different models, depending on the different capabilities of the leader.

In our model, the leader can select the jobs it controls and its power is given by a parameter $0 \leq \alpha \leq 1$, such that it can select any subset of jobs whose total load is at most an α fraction of the total load.

As we show, in some instances, the leader should better exploit its power only partially and let more jobs select their strategy selfishly. Intuitively, this is due to the fact that the leader acts first and its assignment is irrevocable.

The scenarios we propose to study arise in real life applications that provide service to a mixture of independent and controlled clients. For example, several companies suggest computing services on shared machines for private users (using cloud services) as well as local computation tasks. Similarly, in routing problems, some users obey and consult a navigation app, while others don't. The navigation app acts as a leader that determines the decisions (selected path) of some of the clients.

A. Notation and Problem Statement

A load balancing game is given by $G = (\mathcal{J}, \mathcal{M}, \{p_j\} \forall j \in \mathcal{J})$, where \mathcal{J} is a set of n jobs, \mathcal{M} is a set of m machines, p_j is the *weight* (also denoted *size*) of job j . In unweighted games, all the jobs have the same unit size, that is, $\forall j, p_j = 1$.

A Stackelberg load balancing game is given by (G, α) , where G is a load balancing game and $0 \leq \alpha \leq 1$ denotes the maximal fraction of the load controlled by the leader.

Let A be the set of jobs controlled by the leader. Let $P = \sum_{j \in A} p_j$. We have that $\sum_{j \in A} p_j \leq \alpha P$. A profile of a game is a *schedule* $s = \langle s_1, \dots, s_n \rangle \in M^n$ describing the machines on

which the jobs are assigned¹. If $j \in A$ then s_j is determined by the leader and if $j \in \mathcal{J} \setminus A$, then s_j is selected by player j .

For a machine $i \in \mathcal{M}$, the *load* on i in s , denoted $L_i(s)$, is the total size of the jobs assigned on machine i in s , that is, $L_i(s) = \sum_{\{j|s_j=i\}} p_j$. When s is clear from the context, we omit it. It takes p_j time-units to process job j on machine i . As common in the study of job-scheduling games, we assume that all the jobs assigned on the same machine are processed in parallel and have the same cost, defined as the machine's completion time. Formally, the cost of job j in profile s is $C_j(s) = L_{s_j}(s)$. The players that control jobs act selfishly, trying to minimize the cost of the job they control. The leader is trying to optimize some global objective function. Note that if $A = \mathcal{J}$ then all the jobs are controlled by the leader and we have a classical job scheduling problem.

The leader applies its strategy first and determines an assignment for the players in A . Afterwards, the players in $\mathcal{J} \setminus A$ join this partial schedule, each selecting selfishly an assignment for its job. The jobs assigned by the leader cannot change their assignment. The selfish jobs may change their assignment in response to other jobs' assignments.

For a profile s , a job $j \in \mathcal{J} \setminus A$ and a machine $s'_j \neq s_j$, let (s_{-j}, s'_j) denote the profile obtained from s by replacing the strategy of job j by s'_j . That is, the profile resulting from a migration of job j from machine s_j to machine s'_j . A profile s is a *pure Nash equilibrium* (NE) if no selfish job can benefit from unilaterally deviating from its strategy in s to another strategy; i.e., for every job $j \in \mathcal{J} \setminus A$ and every machine s'_j it holds that $C_j(s_{-j}, s'_j) \geq C_j(s)$. The cost of a schedule is defined to be the maximal completion time of a job, also known as *the Makespan* of the schedule, that is, $cost(s) = \max_{j \in \mathcal{J}} C_j(s)$.

It is well known that decentralized decision-making may lead to sub-optimal solutions from the point of view of society as a whole. For a game G , let $S(G)$ be the set of feasible profiles of G . We denote by $OPT(G)$ the cost of a social optimal (SO) solution; i.e., $OPT = \min_{s \in S(G)} cost(s)$. We quantify the inefficiency incurred due to self-interested behavior according to the *price of anarchy* (PoA) [16] and *price of stability* (PoS) [1], [20] measures. The PoA is the worst-case inefficiency of a pure Nash equilibrium, while the PoS measures the best-case inefficiency of a pure Nash equilibrium. Formally,

Definition 1.1: Let \mathcal{G} be a family of games and let G be a game in \mathcal{G} . Let $\Upsilon(G)$ be the set of pure Nash equilibria of the game G . Assume that $\Upsilon(G) \neq \emptyset$.

- The *price of anarchy* of G is the ratio between the *maximal* cost of a NE and the social optimum of G . That is, $PoA(G) = \max_{s \in \Upsilon(G)} cost(s)/OPT(G)$. The *price of anarchy* of the family of games \mathcal{G} is $PoA(\mathcal{G}) = \sup_{G \in \mathcal{G}} PoA(G)$.

We now define the inefficiency measure of a Stackelberg game.

¹In this paper, we only consider *pure* strategies. Unlike mixed strategies, pure strategies may not be random or drawn from a distribution.

Definition 1.2: Let $\langle \mathcal{G}, \alpha \rangle$ be a family of Stackelberg games where \mathcal{G} is a family of games and let G be a game in \mathcal{G} . Let $\Upsilon(G, \alpha)$ be the set of pure Nash equilibria of the game $\langle G, \alpha \rangle$. Assume that $\Upsilon(G, \alpha) \neq \emptyset$.

- The *price of anarchy* of $\langle G, \alpha \rangle$ is the ratio between the *maximal* cost of a NE and the social optimum of G . That is, $PoAL(G, \alpha) = \max_{s \in \Upsilon(G, \alpha)} cost(s)/OPT(G)$. The *price of anarchy* of the family of Stackelberg games $\langle \mathcal{G}, \alpha \rangle$ is $PoAL(\mathcal{G}, \alpha) = \sup_{G \in \mathcal{G}} PoAL(G, \alpha)$.

B. Related Work

The two extreme cases, of $A = \mathcal{J}$ and $A = \emptyset$ are well studied. The case $A = \mathcal{J}$ is the classical minimum makespan problem, while the case $A = \emptyset$ is the classical load balancing game.

The minimum makespan problem is an NP-hard problem and has a rich collection of approximation algorithms. For identical machines, the simple greedy List-scheduling (LS) algorithm [10] provides a $(2 - \frac{1}{m})$ -approximation to this problem. A bit better approximation ratio of $(\frac{4}{3} - \frac{1}{3m})$ is guaranteed by the Longest Processing Time (LPT) algorithm [11]. A PTAS for the minimum makespan problem on identical machines is given in [12]. For related machines (with various speeds), List-scheduling guarantees $\Theta(m)$ -approximation [4] and a PTAS is presented in [6].

Load balancing *game* consist of a set of jobs (players) and a set of machines. Each job is controlled by a selfish agent who aims to minimize his cost - given by the load on the machine it is assigned to ([21]). The concept of the price of anarchy (PoA) was introduced by Koutsoupias and Papadimitriou in [16]. They proved that the price of anarchy of job scheduling games is $2 - \frac{1}{m}$. In [7], Finn and Horowitz presented an upper bound of $2 - \frac{2}{m+1}$ for the price of anarchy in load balancing games with identical machines. Czumaj and Vöcking [5] gave tight bounds for related machines that grow as the number of machines grows.

Other related work consider Stackelberg strategies for resource allocation games with identical players and non-decreasing latency functions which can vary between the resources. In these games, the leader is characterized by the fraction $\alpha \in [0, 1]$ of the players it controls and an optimal allocation can be computed in polynomial time. Our model differs from the previous studies as we do not assume the players to have identical weight or strategy space, hence, the leader is characterized by the specific set of players it controls and calculating a socially optimum assignment is an NP-hard problem.

For routing games on parallel networks for with non-decreasing latency functions, it is known that computing an optimal Stackelberg strategy is NP-hard. There are two known approximation algorithms for the case of splittable (non-atomic) symmetric games with non-decreasing latency functions presented by Roughgarden in [18]; The *Scale* strategy which simply employs the optimal configuration scaled by the fraction of coordinated players. The best bound known for *Scale* PoA is $\frac{4}{3} - \frac{X}{3}$ where $X = \frac{(1-\sqrt{1-\alpha})(3\sqrt{1-\alpha}+1)}{2\sqrt{1-\alpha}+1}$

[13]. The *LLF* (Largest Latency First) strategy presented by Roughgarden in [18] assigns the controlled players to the largest cost strategies in the optimal configuration. The best known upper bound, achieved by Karakostas and Kolliopoulos [13], is equal to $\frac{4}{3}$ for $\alpha \leq \frac{1}{3}$ and $\frac{2(1-\alpha^2)}{2-\alpha-\sqrt{4\alpha-3\alpha^2}}$ for $\alpha > \frac{1}{3}$.

Subsequently, Kumar and Marathe [17] presented a fully polynomial-time approximation scheme for the problem of computing an optimal Stackelberg configuration on parallel links with polynomial latencies. Stackelberg routing in arbitrary networks is studied in [2].

The work mostly related to our study considers games with unsplittable (atomic) flows with non-decreasing latency functions. In these games, different machines may have different latency functions, that are not necessarily linear. The players are identical, but they do have a non-splittable constant size. Thus, the problem of calculating the optimal assignment is polynomially solvable, contrary to our model. Fotakis [8] studied *LLF* and a randomized version of *Scale* and gave upper and lower bounds for them. He also introduced the Stackelberg strategy λ -Cover which assigns to every resource either at least λ or as many coordinated players as the resource has in the social optimum. Finally, he also gave upper bounds for strategies obtained by combining λ -Cover with either *LLF* or *Scale* and upper bounds for games played on parallel links.

Vittorio and Vinci [3] then presented improved bounds for the three Stackelberg strategies studied by Fotakis [8]. For *LLF* they showed *PoA* of exactly $\frac{20-11\alpha}{8}$ for $\alpha \in [0, \frac{4}{7}]$ and $\frac{4-3\alpha+\sqrt{4\alpha-3\alpha^2}}{2}$ for $\alpha \in [\frac{4}{7}, 1]$. For λ -Cover they showed that the *PoA* is for affine functions is $\frac{4\lambda-1}{3\lambda-1}$ and $1 + \frac{4\lambda+1}{4\lambda(2\lambda+1)}$ for linear ones. Finally, for *Scale* they give a bound of $1 + \frac{(1-\alpha)(2h+1)}{(1-\alpha)h^2+\alpha h+1}$, where h is the unique positive integer such that $\alpha \in [r_L(h), r_U(h)]$, with $r_L(h) = \frac{2h^2-3}{2(h^2-1)}$, $r_U(h) = \frac{2h^2+4h-1}{2h(h+2)}$ and $r_L(1) = 0$.

To the best of our knowledge our work is the first to consider a model with unsplittable variable size jobs.

C. Our Results

Let $\langle G, \alpha \rangle$ be a Stackelberg load balancing game. The leader aims to minimize the makespan of the schedule achieved after the addition of selfish jobs. We consider two questions:

- 1) How should the leader choose the jobs it controls?
- 2) How should it schedule them on the machines?

In Section II, we present results for games with two job sizes, that is, for all $j \in \mathcal{J}$, $p_j \in \{1, p\}$. First, in Section II-A we characterize the social optimum in these instances in leader-free settings. In Section II-B, we consider settings where the leader must exploit all of its power and we show that it may result with a worst schedule than a NE in a leader-free environment. In Section II-C we show that when $\alpha \geq \min\{\frac{n_1}{p}, \frac{n_p p}{p}\}$ the leader can guarantee an optimal NE. Furthermore, we show that for every $\alpha \geq 0$, for every game G with two job sizes, $PoAL(G, \alpha) \leq 1 + \frac{p-1}{OPT(G)}$.

In Section III, we analyze the *PoAL* as a function of α and compare it to the *PoA* in a leader-free game. We show that for $\alpha < \frac{1}{m+1}$, $PoAL(\mathcal{G}, \alpha) = PoA(\mathcal{G})$. We also show a lower

bound of $\alpha \geq \frac{1}{m}$ from which we can guarantee $PoAL(\mathcal{G}, \alpha) < PoA(\mathcal{G})$. Furthermore, we present and analyze two strategies for choosing and scheduling the controlled jobs.

In Section IV, we present the leader's heuristics for both choosing the controlled jobs and assigning them to machines. In Section V we present our experimental results.

We conclude in Section VI, where we summarize the work and suggest some directions for future work. Due to space constraints, some of the technical proofs are omitted.

II. INSTANCES WITH TWO JOB SIZES $\{1, p\}$

In this section, we consider the class \mathcal{G}_2 of game instances with only two different job sizes, w.l.o.g., 1 and p . Formally, $G \in \mathcal{G}_2$ if there exists a constant $p > 1$ such that $\forall j \in \mathcal{J}$, $p_j \in \{1, p\}$. We denote by ℓ -job a job of size ℓ . Let n_1 and n_p denote the number of 1-jobs and p -jobs, respectively. It is easy to see that LPT algorithm is optimal for any $G \in \mathcal{G}_2$.

A. Social Optimum in Leader-Free Games

We first provide some simple observations regarding the social optimum and leader-free games in \mathcal{G}_2 .

First, we denote r_G to be the number of machines with exactly $\lceil \frac{n_p}{m} \rceil$ p -jobs on them on a LPT schedule. Thus, $r_G = n_p \bmod m$ when $\frac{n_p}{m} \notin \mathbb{N}$ and there are $m - r_G$ machines with $\lfloor \frac{n_p}{m} \rfloor$ p -jobs. Also, let $h = \lfloor \frac{n_p}{m} \rfloor$, meaning, $n_p = hm + r_G$.

Proposition 2.1: For every load balancing game $G \in \mathcal{G}_2$,

$$OPT(G) = \begin{cases} \lceil \frac{n_p}{m} \rceil p & \text{if } n_1 \leq (m - r_G)p \text{ and } n_p > 0 \\ \lfloor \frac{p}{m} \rfloor & \text{otherwise} \end{cases}$$

Proof: Consider an optimal LPT schedule of $G \in \mathcal{G}_2$. If $n_p > 0$ then LPT first schedules all the p -jobs in a balanced way. The makespan after this stage is $\lceil \frac{n_p}{m} \rceil p$. For the makespan to remain $\lceil \frac{n_p}{m} \rceil p$, the number of 1-jobs must be at most $(m - r_G)p$, since otherwise, $\frac{n_1 + n_p p}{m} > \lceil \frac{n_p}{m} \rceil p$. If $n_1 > (m - r_G)p$, then the number of 1-jobs is sufficient to perfectly balance the load on the machines, up to a gap of 1, hence, the resulting schedule has a makespan of $\lfloor \frac{p}{m} \rfloor$. ■

Proposition 2.2: For every load balancing game $G \in \mathcal{G}_2$, if $PoA(G) > 1$, then in every sub-optimal NE s , every machine a with $L_a(s) > OPT(G)$ processes only p -jobs.

Proof: Let $G \in \mathcal{G}_2$ such that $PoA(G) > 1$. Let s be a NE of G such that $cost(s) > OPT(G)$. Let a be a machine with $L_a(s) > OPT(G)$. By the pigeonhole principle, there must be a machine b for which $L_b < OPT(G)$. Hence, $L_a(s) > L_b(s) + 1$. Machine a processes only p -jobs, as otherwise, any 1-job on machine a would benefit from migrating to machine b , contradicting the stability of s . ■

Next, we characterize instances for which every NE is optimal. First, we calculate some useful values. Denote $G \in \mathcal{G}_2$ to be a game with $n_1 > (m - r_G)p$. In order to calculate the value of $OPT(G)$ we take s to be an LPT schedule which we know is optimal. Each machine has at least h p -jobs on it and there are $m - r_G$ machines with exactly h which has exactly $(m - r_G)p$ 1-jobs scheduled on them since $OPT(G) > \lceil \frac{n_p}{m} \rceil$. Thus, we have n_p p -jobs and $(m - r_G)p$ 1-jobs in order to have all the machines with load of $\lceil \frac{n_p}{m} \rceil$. Hence, there are

more $n_1 - (m - r_G)p$ 1-jobs in \mathcal{J} and the maximal load they reach is $\lceil \frac{n_1 - (m - r_G)p}{m} \rceil$. Denote this value to be B_G and we get in total $OPT(G) = \lceil \frac{P}{m} \rceil = \lceil \frac{n_p}{m} \rceil + B_G$.

For some schedule s' if there is a machine i with $L_i(s') > OPT(G)$ then it must be that there are only p -jobs on i . The number of p -jobs on i is $\lceil \frac{n_p}{m} \rceil + \lfloor \frac{B_G}{p} \rfloor + 1$, meaning, the number of p -jobs that fits in a machine with load $OPT(G)$ and additional 1 p -job. We denote the number of p -jobs on i to be C_G . Using the above definitions, we can characterize instances for which every NE is optimal (proof omitted).

Proposition 2.3: For every load balancing game $G \in \mathcal{G}_2$, $PoA(G) = 1$ iff at least one of the following conditions holds:

- 1) $n_p \leq 1$
- 2) $OPT(G) = \lceil \frac{n_p}{m} \rceil p$
- 3) $OPT(G) > \lceil \frac{n_p}{m} \rceil p$ and $n_1 < (-n_p - C_G) \bmod m - 1$ or $C_G p > \lfloor \frac{(n_p - C_G)p + n_1}{m - 1} \rfloor + p$ or $(n_p p - 1)(m - 1) \leq n_1$

B. The Leader Controls Maximal Load from αP

In this section we assume that the leader must fully exploit its power, that is, the leader must control a subset of jobs whose total size is maximal among subsets of total size at most αP . We assume that the leader has the computational power to identify such a subset. In particular, if there exists a subset of the jobs with total size αP , then the leader must control such a subset. A game in which the leader must control such a subset is denoted $\langle G, = \alpha \rangle$.

The definition of PoAL is adjusted to fit this case as follows:

Definition 2.1: Let $\langle \mathcal{G}, = \alpha \rangle$ be a family of Stackelberg games, and let $G \in \mathcal{G}$. Let $\Upsilon(G, = \alpha)$ be the set of pure Nash equilibria of the game $\langle G, = \alpha \rangle$. Assume that $\Upsilon(G, = \alpha) \neq \emptyset$.

- The *price of anarchy* of $\langle G, = \alpha \rangle$ is the ratio between the maximal cost of a NE and the social optimum of G . That is, $PoAL(G, = \alpha) = \max_{s \in \Upsilon(G, = \alpha)} cost(s) / OPT(G)$. The *price of anarchy* of the family of Stackelberg games $\langle \mathcal{G}, = \alpha \rangle$ is $PoAL(\mathcal{G}, = \alpha) = \sup_{G \in \mathcal{G}} PoAL(G, = \alpha)$.

We first show that controlling the maximal possible load may not always minimize the PoAL.

Proposition 2.4: There exists a game $G \in \mathcal{G}_2$ and $0 \leq \alpha \leq 1$ such that $PoAL(G, = \alpha) > PoA(G)$.

Proof: Let G be a weighted Stackelberg load balancing game with $m = 2$, $p = 4$, $n_p = 3$ and $n_1 = 2$ and let $\alpha = \frac{5}{14}$. Let $\mathcal{J} = \{j_1 \dots j_5\}$ where $p_1 = p_2 = p_3 = 4$ and $p_4 = p_5 = 1$.

We have $P = 14$, thus, the leader must control jobs of total size 5. This implies that the leader controls one 4-job and one 1-job. W.l.o.g. the leader controls jobs j_1 and j_4 . Figure 1(a) presents an optimal schedule of \mathcal{J} . $OPT(G) = 8$ and by Proposition 2.1 $PoA(G) = 1$. We show that $PoAL(G, = \alpha) > 1$. If the leader schedules both of the jobs on the same machine, w.l.o.g., on machine m_1 , then a possible NE is depicted in Figure 1(b). We have $L_1(s) = 9$ and $L_2(s) = 5$. The only job that may benefit from a migration is the 1-job on m_1 . However, this job was assigned by the leader and cannot change its assignment.

If the leader schedules each of the two controlled jobs on a different machine, w.l.o.g., $s_1 = 1$ and $s_4 = 2$, then a possible NE is a one where j_2, j_3 are scheduled on m_1 and j_5 on m_2 . We have, $L_1(s) = 9$ and $L_2(s) = 5$ as depicted in Figure 1(c). The only job that may benefit from a migration is the 1-job on m_1 . However, this job was assigned by the leader and cannot change its assignment.

Hence, $PoAL(G) = \frac{9}{8} > 1 = PoA(G)$. ■

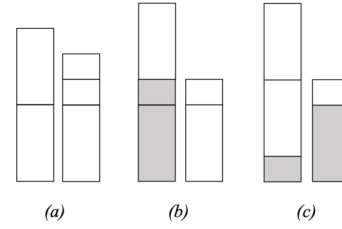


Fig. 1. $PoAL(G) > PoA(G)$. (a) optimal and worst case schedule with only selfish jobs, (b),(c) possible NE profiles with controlled jobs

We show that there is no constant $\alpha < 1$ such that controlling α of the total load is sufficient to guarantee an optimal assignment.

Theorem 2.5: For every $\epsilon > 0$ and every $p > 1$, there exists a load balancing game $G \in \mathcal{G}_2$ such that $PoAL(G, = \alpha) \geq 1 + \frac{1}{p}$ for $\alpha \geq 1 - \epsilon$.

Proof: Given ϵ, p , let $m \geq 3$ be an integer such that $\frac{2p-1}{m} \leq \epsilon$ and let $\epsilon' = \frac{2p-1}{pm}$. Let G be a game on m machines. Let $n_1 = p$ and $n_p = m - 1$. Thus, $P = pm$. Since the leader controls a fraction $1 - \epsilon'$ of the total load and $\epsilon'P = 2p - 1$, it must be that the leader does not control exactly one p -job and $p - 1$ 1-jobs, that is, A consists of a single 1-job and $m - 2$ p -jobs.

In every optimal schedule for G , all the machines have load p . Specifically, on $m - 1$ machines there is one p -jobs and on one machines there are p 1-jobs. Thus, $OPT(G) = \lceil \frac{P}{m} \rceil = p$.

In order to achieve an optimal solution, the leader must schedule the jobs in A in a sub-assignment of an optimal schedule, since otherwise it cannot be completed to an optimal schedule. We show that the leader cannot schedule the jobs of A such that any stable addition of the selfish jobs is optimal.

The only possible assignment for the leader, is to schedule $m - 2$ p -jobs on $m - 2$ machines and the single 1-job on another machine. After the assignment of A , there is an empty machine. A possible stable assignment of the selfish jobs is produced by adding one p -job to the machine with load 1 and assigning the $p - 1$ 1-jobs on the empty machine. (see Figure 2)

In this case, it is easy to see that the resulting schedule is stable against deviations of the selfish jobs, thus, the resulting schedule has cost $p + 1$ and $PoAL = 1 + \frac{1}{p}$. ■



Fig. 2. The only optimal Stackelberg strategy of the controlled (shaded) jobs and a possible non-optimal stable completion of the assignment by the selfish (light) jobs.

C. Games with Optional Control

As shown in section II-B, forcing the leader to fully exploit its power may be harmful. We therefore consider next the more flexible and reasonable setting in which the leader may choose the amount of load it controls out of the maximal allowed fraction α .

Theorem 2.6: For every game $\langle G, \alpha \rangle$ with $G \in \mathcal{G}_2$ and $\alpha \geq \min\{\frac{n_1}{P}, \frac{n_p p}{P}\}$ it holds that $PoAL(G, \alpha) = 1$.

Proof: Let $G \in \mathcal{G}_2$. If $\alpha \geq \frac{n_p p}{P}$, the leader may choose to control all of the p -jobs and none of the 1-jobs. If $\alpha \geq \frac{n_1}{P}$, it can choose to control all of the 1-jobs. The leader would schedule the controlled jobs in a sub-assignment of an optimal schedule. Since all of the selfish jobs have the same size, they will complete the assignment in a balanced schedule, which is optimal. ■

Theorem 2.7: For every $\alpha \geq 0$ and $G \in \mathcal{G}_2$, $PoAL(G, \alpha) \leq 1 + \frac{p-1}{OPT(G)}$.

Proof: Let $G \in \mathcal{G}_2$ be a weighted Stackelberg load balancing game. Assume that the leader assigns the controlled jobs in a sub-assignment of an optimal schedule. We show that for any completion of this sub-assignment to a NE, s , it holds that $cost(s) \leq OPT(G) + p - 1$.

Assume by contradiction that there is a machine m_i with $L_i(s) \geq OPT(G) + p$. By Proposition 2.2 the only selfish job on m_i are p -jobs. Also, since the leader schedule the jobs in a sub-optimal assignment then there is at least one selfish job j on m_i . By the pigeonhole principle there is a machine $m_{i'}$ with $L_{i'} < OPT(G)$. Thus, j would benefit from migrating to $m_{i'}$ and s is not stable. ■

III. INSTANCES WITH ARBITRARY JOB SIZES

Algorithm 1 presents a strategy for the leader, given the job sizes and the fraction of load $0 \leq \alpha < 1$ that the leader may control. Recall that $P = \sum_j p_j$. Thus, the leader may choose to assign any subset of jobs of total load at most αP . We assume that the leader has unlimited computational power, in particular, it may solve NP-hard problems.

Algorithm 1 - A strategy for assigning the controlled jobs on m machines

- 1: Sort the jobs such that $p_1 \geq \dots \geq p_n$
- 2: Let k be the maximal index for which $\sum_{j \leq k} p_j \leq \alpha P$.
- 3: Assign the k first jobs in a way that minimizes the maximal load on a machine.

Claim 3.1: Algorithm 1 gives a $(1 + \frac{1 - \frac{1}{m}}{1 + \lfloor \frac{k}{m} \rfloor})$ -approximation.

Proof: The analysis is similar to the analysis of the PTAS for the minimum makespan problem [9]. Let s' be the schedule

of the k longest jobs and let s denote the NE profile after the selfish jobs join. If $cost(s') = cost(s)$ then s is optimal for G . Otherwise, $cost(s) \geq OPT(G)$.

Let j be the job determining $cost(s)$. Since s is NE, every machine has load of at least $cost(s) - p_j$. Therefore, $P \geq m(cost(s) - p_j) + p_j$. Also, since the jobs are sorted in nonincreasing order of processing times, we have that $p_j \leq p_{k+1}$ and therefore, $P \geq m \cdot cost(s) - (m-1)p_{k+1}$. Furthermore, a lower bound for the optimal solution is a perfectly balanced schedule, thus, $OPT(G) \geq \frac{P}{m}$, which implies that $cost(s) \leq OPT(G) + (1 - \frac{1}{m})p_{k+1}$.

Next we bound p_{k+1} in terms of $OPT(G)$. To obtain such a bound, consider the $k+1$ longest jobs. In an optimal schedule, some machine is assigned at least $\lceil \frac{k+1}{m} \rceil \geq 1 + \lfloor \frac{k}{m} \rfloor$ of these jobs. Since each of these jobs has processing time at least p_{k+1} , we conclude that $OPT(G) \geq (1 + \lfloor \frac{k}{m} \rfloor)p_{k+1}$, which implies that $p_{k+1} \leq \frac{OPT(G)}{1 + \lfloor \frac{k}{m} \rfloor}$ and finally,

$$cost(s) \leq OPT(G) \left(1 + \frac{1 - \frac{1}{m}}{1 + \lfloor \frac{k}{m} \rfloor}\right)$$

Our next results identify the threshold fraction α , such that controlling less than fraction α may not be helpful at all while controlling at least fraction α is guaranteed to be beneficial.

Theorem 3.2: If $\alpha < \frac{1}{m+1}$ then $PoAL(G, \alpha) = PoA(G)$.

Proof: Recall that $PoA(G) = 2 - \frac{2}{m+1}$. We show that there exists a game G for which $PoAL(G, \alpha) = 2 - \frac{2}{m+1}$ for every $\alpha < \frac{1}{m+1}$.

Given m , the set of players in G consists of $m^2 - m$ jobs of size 1 and two jobs of size m . Thus, $P = m^2 + m$. $OPT(G) = m + 1$ achieved by assigning the two long jobs on different machines and balance the load with the unit jobs.

Assume the leader controls a fraction $\alpha < \frac{1}{m+1}$ of the load. Thus, it controls load less than m , implying that it controls at most $m-1$ unit jobs. Therefore, after the leader schedules the controlled jobs, there is an empty machine m_1 .

In a possible NE schedule s the two m -jobs are assigned on m_1 and all of the unit jobs are balanced on the remaining machines, each having load m .

We show that s is NE. Clearly, none of the unit jobs on machine $i \neq 1$ may benefit from a migration since $L_i(s) = m$ and $L_1(s) = 2m$. Moreover, since for every $i \neq 1$ $L_i(s) = m$ and there are only two m -jobs scheduled on m_1 , they will not benefit from a migration.

Therefore, for any $\alpha < \frac{1}{m+1}$ we have a NE s with $cost(s) = 2m$. Thus, $PoAL(G, \alpha) = \frac{2m}{m+1}$. ■

In order to show that controlling a fraction $\alpha \geq \frac{1}{m-1}$ guarantees a reduced equilibrium inefficiency, we characterize instances that achieve the worst PoA. We first show that every optimal solution must be perfectly balanced and then that there are at least m relatively short jobs.

Claim 3.3: For every game G with $PoA(G) = \frac{2m}{m+1}$, $OPT(G) = \frac{P}{m}$.

Proof: Let G be a game with $PoA(G) = \frac{2m}{m+1}$ and let $r = \frac{OPT(G)}{m+1}$. Let s be a schedule such that $cost(s) = 2mr$. Therefore, there is a machine M_a with $L_a(s) = 2mr$.

Assume by contradiction that $OPT(G) > \frac{P}{m}$, thus, $P < r(m^2 + m)$ then $\sum_{i \neq a} L_i(s) < r(m^2 - m)$. Therefore, there is a machine $b \neq a$ with $L_b < rm$. Also, since $L_a(s) = 2mr$ and $OPT(G) = r(m+1)$ it must be that M_a processes at least two jobs and the shortest job on M_a has size at most rm . By migrating to M_b , the shortest job on M_a will reduce its cost below $2rm$, contradicting the stability of s . ■

Theorem 3.4: If $\alpha \geq \frac{1}{m}$ then $PoAL(\mathcal{G}, \alpha) < 2 - \frac{2}{m+1}$.

Theorem 3.5: If the leader controls m jobs, $PoAL(\mathcal{G}) < PoA(G)$.

Proof: Let G be a load balancing game with $PoA(G) = 2 - \frac{2}{m+1}$ and $m \geq 3$ and let s be the schedule for which $cost(s) = (2 - \frac{2}{m+1})OPT(G)$. Denote i to be a machine with $L_i(s) = cost(s)$. Also, $L_i(s) - OPT(G) = (2 - \frac{2}{m+1})OPT(G) - OPT(G) = (1 - \frac{2}{m+1})OPT(G)$ and for $m \geq 3$ we get $L_i(s) - OPT(G) \geq \frac{OPT(G)}{2}$. Since $L_i(s) > OPT(G)$ there must be a machine i' with $L_{i'}(s) < OPT(G)$. Therefore, i contains two jobs with $p_j > \frac{OPT(G)}{2}$, since otherwise, s is not NE.

Moreover, there are at most m jobs with $p_j > \frac{OPT(G)}{2}$ since otherwise, for every schedule s' there must be a machine i' with $L_{i'}(s') > OPT(G)$ and the optimal value of G can not be achieved. Contradicting the definition of $OPT(G)$.

Let A be the group of jobs with $p_j > \frac{OPT(G)}{2}$. If the leader controls A , since $|A| \leq m$, it may schedule each job on a different machine. Let s' be the resulted schedule s' after the selfish jobs reach NE.

We show that the schedule s' is a NE for a game with all selfish jobs. Assume there is a job j scheduled on machine a that may benefit from a migration. Then since all of the jobs with $p \leq \frac{opt(G)}{2}$ are satisfied, it must be that $p_j > \frac{OPT(G)}{2}$ and there is a machine b with $L_b(s') < \frac{OPT(G)}{2}$. Also, machine b has an additional job j' scheduled on it, since otherwise j is the only job on a and it may not benefit from a migration. It must be that $p_{j'} \leq \frac{OPT(G)}{2}$ since the leader scheduled all of the larger jobs on distinct machines. Thus, j' may benefit by migrating to b . Contradicting the stability of the jobs with sizes smaller than $\frac{OPT(G)}{2}$.

Since s' is a NE then it must be that $cost(s') \leq (2 - \frac{2}{m+1})OPT(G)$. Also, we showed that in every NE with cost $(2 - \frac{2}{m+1})OPT(G)$ it must be that there is a machine with two jobs larger than $\frac{OPT(G)}{2}$. Thus, since s' does not satisfy the condition, $cost(s') < (2 - \frac{2}{m+1})OPT(G)$. ■

Algorithm 2 presents a strategy for the leader, given a fraction α , which is the load of jobs the leader control.

Algorithm 2 - A strategy for finding an approximation for a schedule with m machines

- 1: Sort the jobs such that $p_1 \geq \dots \geq p_n$
 - 2: Assign the k first jobs such that $\sum_{i=1}^k p_i \leq \alpha P$ in First-Fit with limit of C^* .
-

Theorem 3.6: For every $\alpha \leq \frac{m}{m+1}$, Algorithm 2 gives $PoAL(\mathcal{G}, \alpha) \geq 2 - \frac{2}{m+1}$.

Proof: Given m , let G be a game for which the set of players in G consists of m jobs of size m and m jobs of size 1. Thus, $OPT(G) = m+1$ is achieved in a balanced schedule and $PoA(G) = 2 - \frac{2}{m+1}$. Let $\alpha \leq \frac{m}{m+1}$ be the fraction of controlled load. Therefore, the leader controls at most $m-1$ jobs with size m .

If $C^* \geq 2m$, if the leader controls only one job, then the cost after the selfish jobs join may be the worst case. Otherwise, the leader assigns at least two m -jobs on a machine. in both case $cost(s) \geq 2m$ and $PoAL(G, \alpha) \geq 2 - \frac{2}{m+1}$.

If $C^* < 2m$ the leader assigns only one m -job on at most $m-1$ machines. Let M_1 be one of those machines. In a possible NE schedule s a selfish m -job is assigned on M_1 , the selfish m -jobs are assigned on different machines and all of the unit jobs are assigned on a single machine. Therefore, each machine having load m except M_1 , with $L_1(s) = 2m$. Thus, $PoAL(G, \alpha) = 2 - \frac{2}{m+1}$. ■

IV. OUR HEURISTICS

In this section we describe the heuristics we have designed and implemented. An instance in our experiments is characterized by a set of jobs, a number of machines and two different heuristics. The first defines the leader's strategy and depends on the fraction of load it controls and the second defines the selfish players behaviour. We describe each of these two classes of heuristics separately.

A. Leader's Strategy

A leader's strategy consists of two steps: (i) Choosing the controlled jobs, (ii) Scheduling the controlled jobs. In this section we describe the heuristics we propose for these steps. Note that the leader's strategy is independent of the selfish jobs' behaviour, however, the leader may benefit and adjust its heuristic if it is known in advance how the selfish jobs will act after it is done assigning the controlled jobs.

1) *Choosing the Controlled Jobs:* Given a fraction $0 \leq \alpha \leq 1$, the leader needs to choose the jobs it controls. Recall that $P = \sum_j p_j$. Given $0 \leq \alpha \leq 1$, the leader may control jobs of total load at most αP . We implemented three simple heuristics. In the first heuristic, denoted *Pick Smallest*, the leader sorts the jobs in non-decreasing order of size and adds jobs according to this order as long as their total size is at most αP .

In the second heuristic, the leader sorts the jobs in non-increasing order. For choosing the controlled jobs there are two options: it may choose the largest jobs prefix of this sorted set whose total size is at most αP and stop when the next job can not fit, we denote this heuristic *Pick Largest - Stop*, or it may add jobs according to the sorted order and skip jobs whose addition will make the total load more than αP , we denote this strategy *Pick Largest - Skip*.

For example, assume the jobs sizes are $\{1, 2, 3, 4\}$ and let $\alpha = \frac{1}{2}$. Using *Pick Smallest*, the leader controls jobs of sizes $\{1, 2\}$, in *Pick Largest - Stop*, it controls only the job of size 4 and in *Pick Largest - Skip* it controls jobs of sizes $\{4, 1\}$.

2) *Leader's Scheduling Strategy*: Given the set of controlled jobs, the leader's next mission is to schedule them on the machines. We assumed the leader has limited computational power, meaning, it cannot solve *NP*-hard problems and in particular, it cannot calculate an optimal schedule for the jobs.

The first algorithm we implemented returns an LPT schedule. Meaning, the leader first sorts the controlled jobs in a non-increasing order. Next, it assigns one job at a time to a least loaded machine. We denote this heuristic *Leader's LPT*.

The second algorithm uses the bin packing algorithm *First Fit*. We denote this heuristic *Leader's FF*. The bin packing problem is an optimization problem, in which items of different sizes must be packed into a finite number of bins or containers, each of a fixed given capacity, in a way that minimizes the number of bins used. First Fit algorithm packs each item into the first bin where it fits, possibly opening a new bin if the item cannot fit into any currently open bin.

We consider the machines as bins and the jobs as items. Let $L = \frac{\sum_{j \in \mathcal{J}} p_j}{m}$. Clearly, L is a lower bound on *OPT*. In our heuristics, the bins' capacity is determined to be γL for different values of $\gamma \in \{0.9, 1, 1.1\}$. The leader sorts the job on non-increasing order and then uses First Fit with the chosen bins capacity to schedule the controlled jobs on the machines. If some job cannot fit into any machine without an overflow beyond γL (this may happen if α is relatively large), then the jobs is assigned on a lightly loaded machine.

B. Selfish Player's Strategies

The selfish players behaviour also has significant impact on the results. We considered two methods according to which the selfish players reach a NE.

In the first method the selfish players are added to the game sequentially, each added to a least loaded machine at that time.

The jobs are considered in LPT order. We denote this strategy *Player's LPT*. We claim that in this case, after adding the selfish players, the schedule is stable against deviation of the selfish players. The proof of the following claim follows from the analysis of LPT for classical load balancing games [21].

Claim 4.1: LPT algorithm for non-empty machines produces a NE for the selfish jobs.

In the second method the jobs are added greedily in arbitrary order and then we use *Best Response Dynamics (BRD)*. The best response dynamics is one of the most elementary methods in game theory. Using Round Robin according to the jobs assignment order. Each player at its turn tries to improve its state. If there is a machine for which the player may benefit from a migration, it will migrate to a most beneficial option. Otherwise, it stays on the same machine. The algorithm stops when non of the players has a beneficial migration. It is known that BRD algorithm converges to a NE when starting the algorithm with empty machines. The discussed case of adding players to non-empty machines is a sub-problem of it and achieves NE similarly.

V. EXPERIMENTAL RESULTS

Our instances consist of 20 machines and 100 jobs. It is common to use exponential times for job processing times as it is often a good approximation of service times [19]. In order to work with integers, we used geometrical distribution and chose parameter 0.13 which provides instances with mostly small jobs, with an amount of larger jobs that allows many combinations of equilibrium with different makespan values. Diversity of job sizes and relatively low loads, enables a good comparison between different heuristics.

The random jobs generator may create jobs whose size is larger than the optimal makespan, calculated by $\frac{\sum_{j \in \mathcal{J}} p_j}{m}$. We address these jobs as outliers. The presence of such jobs hurts the comparison of the different heuristics, since they may cause that an instance will have the same makespan for all strategies. By scheduling one such long job on a machine, non of the selfish jobs may benefit by choosing this machine. In order to emphasis the differences between the different heuristics, we remove the outliers using the following method. We note that on average 1 – 2 jobs were removed from each instance.

Algorithm 3 - Removing outliers

- 1: Calculate $avgLoad(G) = \frac{\sum_{j \in \mathcal{J}} p_j}{m}$
 - 2: Remove every job $j \in \mathcal{J}$ with $p_j > avgLoad(G)$.
 - 3: Repeat steps 1, 2 while some job was removed.
-

In the following sections we present our experimental results. These results were obtained by running our heuristics on the same 100 instances, each consisting of 100 players and 20 machines. For each instance we calculated the lower bound, $L = \sum_j p_j / M$, on the optimal makespan. For each experiment we present in the diagrams the average makespan scaled by L . For example, if the average makespan in all the runs performed in some experiment is $1.2L$ then the corresponding bar in the figure has height 1.2. Since $L \leq OPT$, this ratio is an upper bound on the price of anarchy.

A. Choosing the Controlled Jobs

In this section, we present our results for the comparison of the first part in the leader's strategy. Specifically, we compare how the performance is affected by the fraction of load controlled by the leader and the different heuristics for choosing the leader's controlled jobs. We consider all the possible combinations of leader and selfish players strategies. Recall that *Leader's FF* algorithm schedules each job on the first machine where it fits with limit of $\gamma \frac{\sum_{j \in \mathcal{J}} p_j}{m}$ on the totals' machine load for $\gamma \in \{0.9, 1, 1.1\}$, possibly starting to schedule on a new machine if the job cannot fit into any currently open machine. In the experiment described below, we fixed $\gamma = 1$ as a representative case for the strategy.

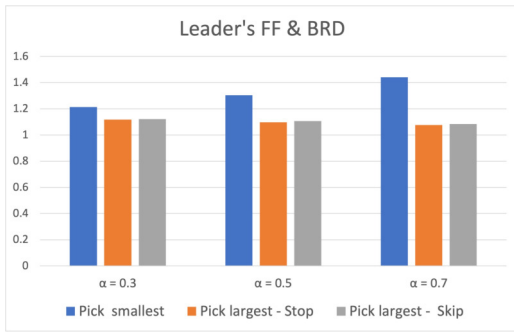


Fig. 3. Comparing jobs choosing strategies for Leader's FF with BRD

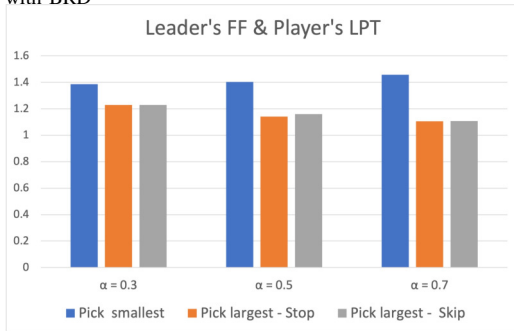


Fig. 4. Comparing jobs choosing strategies for Leader's FF with Player's LPT

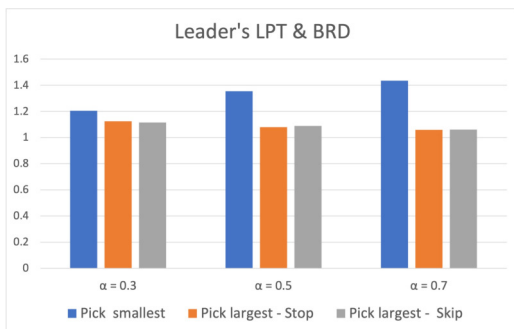


Fig. 5. Comparing jobs choosing strategies for Leader's LPT with BRD

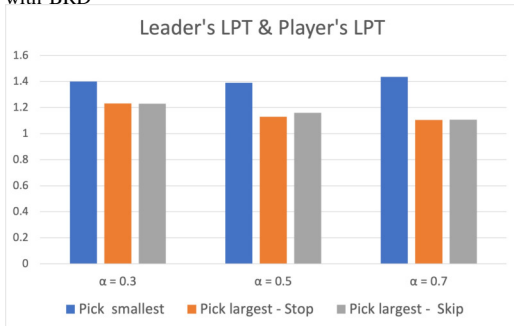


Fig. 6. Comparing jobs choosing strategies for Leader's LPT with Player's LPT

Figures 3,4,5 and 6 present the differences between the

possible leader's strategies for choosing controlled jobs for every possible combination of considered strategies for the leader and selfish players. The figures present the PoA (y axis) as a function of α for every strategy.

First, we can conclude that *Pick Smallest* provides the worst results for all α values and heuristics. As shown in the figures, the PoA increases with α , meaning that the leader's control hurts the social cost.

In contrast, we can also conclude that for the other two strategies *Pick Largest - Skip* and *Pick Largest - Stop*, as α increases, the final schedule is better and the leader reduces the makespan in average by factor 1.16, 1.11, 1.08 if it controls 0.3, 0.5, 0.7 fraction of the load respectively. Other than that, the difference between both strategies is minor, with a slight advantage to the *Pick Largest - Stop* strategy in most of the cases.

B. Leader's Scheduling Strategy

In this section we discuss the optimal strategy we recommend the leader to apply on the controlled jobs. The best strategy may differ between the amount of load the leader controls, the selfish players behaviour or other parameters we examine.

The experiments that consider the selection of the controlled jobs, reported in section 5.1, reveal that *Pick Largest - Stop* has the best performance, thus, we use it in the following experiments.

1) *Leader's First Fit*: Our next experiments analyze the influence of the parameter γ in the *Leader's FF* strategy. Recall that the *Leader's FF* strategy schedules the controlled jobs using *First Fit* algorithm with machine capacity of $\gamma \sum_j \frac{p_j}{m}$ for $\gamma > 0$. We applied this strategy with $\gamma \in \{0.9, 1, 1.1\}$. These values were chosen since we aim to have approximate PoA closer to 1; if we choose a low γ value, the selfish players may determine the makespan since larger jobs may join non-empty machines.

If we choose higher γ value, then already the jobs assigned by the leader may cause a high makespan. On the other hand, this leaves more empty machines for the selfish jobs and potentially prevents them from reaching load higher than $\gamma \sum_j \frac{p_j}{m}$ on the remaining machines.

Figures 8 and 7 present the approximated PoA with regards to the discussed options for γ parameter and the controlled fraction α achieved with $\alpha \in \{0.3, 0.5, 0.7\}$ and $\gamma \in \{0.9, 1, 1.1\}$. In the experiments described in Figure 7 the selfish jobs join and perform BRD. In the experiments described in Figure 8 they are added using LPT rule.

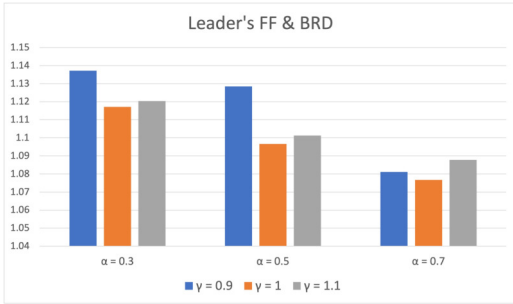


Fig. 7. Comparing γ parameter for *Leader's FF* strategy with *BRD*

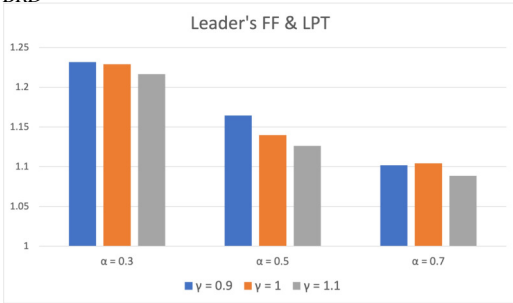


Fig. 8. Comparing γ parameter for *Leader's FF* strategy with *Selfish LPT*

As presented in Figure 7, when the selfish players use *Player's BRD* strategy, for every fraction α , the best performance is achieved when $\gamma = 1$. For $\gamma = 0.9$ we have significantly worst result for $\alpha \in \{0.3, 0.5\}$, but better than $\gamma = 1.1$ for $\alpha = 0.7$. This difference is explainable since the leader have the most significant control for this α and both approximated PoA results values are below 1.1 and for $\gamma = 1.1$ the leader schedules as much as it can near the load of $1.1 \sum_j \frac{p_j}{m}$ which ensured result near 1.1. In contrast to $\gamma = 0.9$ which allowed lower results.

On the other hand, for *Player's LPT* in Figure 8 we get consistent results for all α values. We get the worst results for the value $\gamma = 0.9$. Next, we have $\gamma \in \{1, 1.1\}$ with close results, with a minor advantage to $\gamma = 1.1$.

2) *Leader's Strategy*: We conclude with experiments that compare the leader's strategy choice with regards to the selfish players behaviour. Based on the results of these previous experiments, in all the experiments we report in this section, the controlled jobs are chosen using *Pick Largest - Stop* and for *Leader's FF* we use parameter $\gamma = 1$. In the experiments presented bellow, we compared which leader's strategy gives better results with respect to the applied player's strategy for every α choice.



Fig. 9. Comparing leader's possible strategies for *Player's BRD*

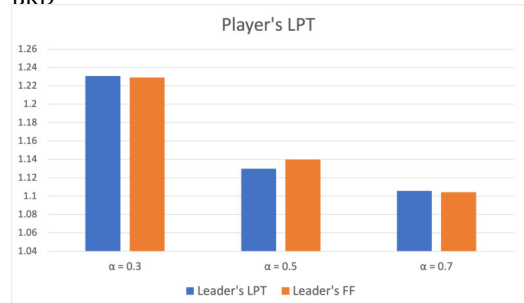


Fig. 10. Comparing leader's possible strategies for *Player's LPT*

The results shown in both Figures 9 and 10, imply that for $\alpha = 0.3$ we get better makespan when using *Leader's FF* strategy. On the other hand, for larger α we get that *Leader's LPT* perform better. Moreover, we can conclude that when the selfish players use BRD, the difference between the makespans resulting from the two leader's strategy are more significant comparing to games in which the selfish players are added by LPT.

Another interesting result we can infer from these experiments, is the difference between the results for selfish players behaviour. For all α values and leader's strategy choice, we get much lower makespans when the selfish players use BRD.

VI. CONCLUSIONS

In this work we examined the potential influence of a leader who control a subset of the jobs in load balancing games. The leader assigns the jobs it controls and the other jobs then select their assignments in a selfish way. We present theoretical as well as experimental results answering several questions regarding the advantages and disadvantages of controlling a fraction of the load. We have designed and implemented several heuristics for possible strategies the leader may apply when choosing the jobs to control and when scheduling them on the machines.

In the theoretical results, we proved that if the leader is obligated to control a maximal amount of the given fraction, the resulting schedule may be worse than a NE schedule in a leader-free environment. On the other hand, for a setting in which the leader may choose to exploit only part of its power, we presented tight bounds on the fraction α of the controlled load that may improve the game cost. Additionally,

we presented several relations between the common $PoA(\mathcal{G})$ measure and the $PoAL(\mathcal{G}, \alpha)$ measure which we defined.

Our experiments show that controlling the largest jobs in a game is the best strategy for the leader, while controlling the smallest jobs may cause significant damage. In fact controlling no job may be better than controlling only small ones. One of the main conclusions that resulted from the experiments is that controlling a larger fraction of the load improves the game cost when controlling the larger jobs.

We have observed that when using the heuristic *Leader's FF*, the performance is significantly influenced by the choice of the capacity to which the machines are loaded. Let $L = \sum_j \frac{p_i}{m}$, then for a parameter γ , the leader adds jobs to machine in a First-Fit matter, where each job is added to the first machine whose addition will result in total load at most γL . In general, the best result achieved for this strategy is when $\gamma = 1$. Although, if the selfish players scheduling strategy is known to be LPT, then filling the machines up to capacity $1.1L$ results is slightly better outcomes. Also, for large α values the results for all γ values are very close, but on the other hand for low α we can definitely conclude that we should not use γ less than 1. Future work on this algorithm may find more characteristics for choosing the γ value.

Another conclusion we can infer, is that knowing the selfish player's behaviour in advance may be helpful for the leader. Similarly, the strategy choice depends also on α - the fraction of controlled jobs. In general, for both selfish players methods it holds that the *Leader's LPT* strategy gives the best results, but if it is known that the selfish players acts in BRD and the controlled fraction is low we may consider using *Leader's FF*.

For future work it may be interesting to improve the *Leader's FF* strategy and characterize the jobs that the leader should choose specifically for using this heuristic. *Leader's FF* is a heuristic in which the leader assigns the controlled jobs on a subset of the machines and leave some of the machines empty for the selfish jobs. We believe that additional algorithms using this approach should be considered and analyze and may perform even better than *Leader's FF*. Also, it may be interesting to get inspiration from additional known approximation algorithms for the minimum makespan problem as we did with *Leader's LPT* which is based on the algorithm presented in [7].

In the theoretical side, we believe that there are many interesting open problems in this setting. For example, analyzing weighted Stackelberg games of congestion games with non-singelton strategy space, non symmetric singelton games, or games with non-uniform resources, that is, machines with different speeds. Also, in our settings the leader has unlimited computational power, meaning, it is able to solve NP-hard problems. It will be interesting to have results for a leader with a limited computational power. Another interesting direction

is to analyze the consequences of controlling a specific given subset of jobs $A \subseteq \mathcal{J}$ instead of a fraction α of the load.

REFERENCES

- [1] E. Anshelevich, A. Dasgupta, J. Kleinberg, E. Tardos, T. Wexler, and T. Roughgarden. The price of stability for network design with fair cost allocation. *SIAM Journal on Computing*, 38(4):1602–1623, 2008.
- [2] V. Bonifaci, T. Harks, and G. Schäfer. Stackelberg Routing in Arbitrary Networks. *Mathematics of Operations Research*, 35(2), 330–346, 2010.
- [3] V. Bilò and C. Vinci. On stackelberg strategies in affine congestion games. *Theory of Computing Systems*, 63(6):1228–1249, 2019.
- [4] Y. Cho and S. Sahni. Bounds for list schedules on uniform processors. *SIAM Journal on Computing*, 9(1):91–103, 1980.
- [5] A. Czumaj and B. Vöcking. Tight bounds for worst-case equilibria. *ACM Trans. Algorithms*, 3(1):4:1–4:17, 2007.
- [6] L. Epstein and J. Sgall. Approximation schemes for scheduling on uniformly related and identical parallel machines. In *Proc. of the 7th European Symposium on Algorithms(ESA)*, 1999.
- [7] G. Finn and E. Horowitz. A linear time approximation algorithm for multiprocessor scheduling. *BIT Numerical Mathematics*, 19(3):312–320, 1979.
- [8] D. Fotakis. Stackelberg strategies for atomic congestion games. *Theory of Computing Systems*, 47(1):218–249, 2010.
- [9] M.R. Garey and R.L. Graham. Bounds for Multiprocessor Scheduling with Resource Constraints. *SIAM J. J. Comput.*, 4(2): 187–200, 1975.
- [10] R.L. Graham. Bounds for certain multiprocessing anomalies. *Bell Systems Technical Journal*, 45:1563–1581, 1966.
- [11] R.L. Graham. Bounds on multiprocessing timing anomalies. *SIAM J. Appl. Math.*, 17:263–269, 1969.
- [12] D.S. Hochbaum and D.B. Shmoys. Using dual approximation algorithms for scheduling problems: Practical and theoretical results. *Journal of the ACM*, 34(1):144–162, 1987.
- [13] S. G Karakostas, G. Kolliopoulos. Stackelberg strategies for selfish routing in general multicommodity networks. *Algorithmica*, 2009.
- [14] H. Kellerer and U. Pferschy. Cardinality constrained bin-packing problems. *Annals of Operations Research*, 92:335–349, 1999.
- [15] E. Koutsoupias and C. Papadimitriou. Worst-case equilibria. *STACS*, pages 404–413, 1999
- [16] E. Koutsoupias and C. Papadimitriou. Worst-case equilibria. *Computer Science Review*, 3(2):65–69, 2009.
- [17] VS A. Kumar and M. V Marathe. Improved results for stackelberg scheduling strategies. In *International Colloquium on Automata, Languages, and Programming (ICALP)*, 2002.
- [18] T. Roughgarden. Stackelberg scheduling strategies. *SIAM Journal on Computing*, 33(2):332–350, 2004.
- [19] P. M. Swamidass (Eds.), Exponential service times. In *Encyclopedia of Production and Manufacturing Management*, 2000.
- [20] A. S. Schulz and N. E Stier Moses. On the performance of user equilibria in traffic networks. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '03*, pages 86–87, 2003.
- [21] B. Vöcking. *Algorithmic Game Theory*, chapter 20: Selfish Load Balancing. Cambridge University Press, 2007.

Intuitionistic Fuzzy Model of the Hungarian Algorithm for the Salesman Problem and Software Analysis of a Shipping Company Example

Velichka Traneva

Prof. Asen Zlatarov University
 1 Prof. Yakimov Blvd, Burgas 8000, Bulgaria
 Email: veleka13@gmail.com

Deyan Mavrov

Prof. Asen Zlatarov University
 1 Prof. Yakimov Blvd, Burgas 8000, Bulgaria
 Email: dg@mavrov.eu

Stoyan Tranev

Prof. Asen Zlatarov University
 1 Prof. Yakimov Blvd, Burgas 8000, Bulgaria
 Email: tranev@abv.bg

Abstract—Here we propose for the first time a temporal intuitionistic fuzzy extension of the Hungarian method for solving the Travelling Salesman Problem (TIFHA-TSP) based on intuitionistic fuzzy logic and index matrices theories. The time for passing a given route between the settlements depends on different factors. The expert approach is used to determine the intuitionistic fuzzy time values for passing the routes between the settlements. The rating coefficients of the experts take the times into account. We are also developing an application for the algorithm’s provision to use it on a real case of TIFHA-TSP.

I. INTRODUCTION

HAMILTON [11] defined the travelling salesman problem (TSP). The Hungarian algorithm (HA) was described by Kuhn in 1955 [6]. In today’s dynamic environment, some of the parameters of this problem are unclear and rapidly changing. Traditional optimisation methods are not easily applicable to this problem. The use of intuitionistic fuzzy logic proposed by Atanassov [8] as an extension of Zadeh’s fuzzy (F) logic [12] allows us to model uncertainty. The TSP and related problems have been solved through various techniques, including meta-heuristic algorithms [1]. FTSP with trapezoidal and triangular fuzzy costs has been solved in [2], [14] using the fuzzy Hungarian method. An improved Hungarian method is described in [15] for FTSP. The optimal solution of triangular intuitionistic FTSP is found [10] using the fuzzy HA. From the literature review we can see that optimal solutions in uncertainty are found only with triangular IF parameters, which are a special case of IF sets. Here we will extend our previous implementation of an IF Hungarian method [16], [19] by using temporal IF pairs and index matrices [9] to find the solution for the TSP at a certain moment in time to fulfil delivery requests. We will denote this approach as Temporal IF HA for the TSP (TIFHA-TSP). We are also in the process of developing a software application for the provision of TIFHA-

TSP and with its help we apply TIFHA-TSP in a real case. The experts’ ratings are taken into account in a way similar to [8]. The rest of this work contains the following sections: Section II describes the definitions of index matrices (IMs) and temporal intuitionistic fuzzy pairs (TIFPs). In Section III, we describe TIFHA-TSP and software for its implementation, and then apply it to a real case. Section IV marks some conclusions and some aspects for future research.

II. TIFP AND INDEX MATRICES

In this section, we will recall the definitions of temporal intuitionistic fuzzy pairs (TIFPs) and index matrices (IMs), as well as some operations and relations with them from [3], [9].

2.1. Temporal Intuitionistic Fuzzy Pair (TIFP)

Let $T = \{t_1, \dots, t_g, \dots, t_f\}$ be a fixed time-scale. A TIFP has the form of $\langle a(t), b(t) \rangle$, where $a, b: T \rightarrow [0, 1]$ and $a(t) + b(t) \leq 1$ for $t \in T$. With two TIFPs $x = \langle a, b \rangle$ and $y = \langle c, d \rangle$ there are different definitions of conjunction and disjunction [3], [16]. We will modify subtraction this way:

$$x(t) - y(t) = \begin{cases} \langle 0, 1 \rangle & \text{if } a = b \text{ \& } c = d \\ \langle a, b \rangle & \text{if } c = 0 \text{ \& } d = 1 \\ \langle \max(0, a - c), \min(1, b + d, 1 - a + c) \rangle & \text{otherwise} \end{cases}$$

Let $R_{\langle a(t), b(t) \rangle} = 0.5(2 - a(t) - b(t))(1 - a(t))$ following [5]. Then, as per [8], [19]:

$$x(t) \geq_R y(t) \text{ iff } R_{\langle a(t), b(t) \rangle} \leq R_{\langle c(t), d(t) \rangle}. \quad (1)$$

A TIFP x is named as “temporal intuitionistic fuzzy false pair” (TIFFalseP) if and only if $\inf a(t) \leq b(t)$, while x is named as “false pair” (TFalseP) iff $a(t) = 0, b(t) = 1$.

2.2. Temporal Intuitionistic Fuzzy Index Matrix (TIFIM)

We remind the definition of a TIFIM [9]

$$A(T) = [K, L, T, \{\langle \mu_{k_i, l_j, t_g}, \nu_{k_i, l_j, t_g} \rangle\}],$$

$t_g \in T$	l_1	...	l_n
k_1	$\langle \mu_{k_1, l_1, t_g}, \nu_{k_1, l_1, t_g} \rangle$...	$\langle \mu_{k_1, l_n, t_g}, \nu_{k_1, l_n, t_g} \rangle$
\vdots	\vdots	...	\vdots
k_m	$\langle \mu_{k_m, l_1, t_g}, \nu_{k_m, l_1, t_g} \rangle$...	$\langle \mu_{k_m, l_n, t_g}, \nu_{k_m, l_n, t_g} \rangle$

(2)

where \mathcal{S} be a fixed set of indices, $(K, L, T \subset \mathcal{S})$, and its elements are TIFP. T is a some fixed temporal scale and its element $t_g (g = 1, \dots, g, \dots, f)$ are time-moments.

Work is supported by the Asen Zlatarov University through project Ref. No. NIX-449/2021 “Index matrices as a tool for knowledge extraction”.

The operations with TIFIMs, such as transposition, addition, termwise multiplication, projection, substitution, internal subtraction, aggregated global internal operation, aggregation and index type operations, are given in [9], [17], [18], [19].

III. AN OPTIMAL TEMPORAL IF HA FOR THE TSP

This section describes a type of temporal intuitionistic fuzzy TSP to find the the fastest route to fulfilling all delivery requests to certain sites with temporal intuitionistic fuzzy data. The IFTSP has the following description:

The seller must visit m settlements $K = \{k_1, \dots, k_i, \dots, k_m\}$ at time t_f . He wants to start from a certain settlement, visit each settlement once and then return to his starting point. The time c_{k_i, l_j, t_f} (for $1 \leq i, j \leq m, i \neq j$) for switching from settlement k_i to l_j at a particular moment t_f are given TIFPs, depending on the peak hours of the day, the condition of the roads, the road conditions, etc. factors. The ratings of the experts $\{r_1, \dots, r_s, \dots, r_D\}$ in the form of TIFPs are defined on the basis of their participation in $\lambda_s(t_f) (s = 1, \dots, D)$ evaluating time procedures respectively before a particular moment t_f and were reflected in the final assessment of the travel time from one point to another. Our aim is to choose the visiting sequence of the settlements so that its total time is minimal.

A. An Optimal Temporal Intuitionistic Fuzzy HA for TSP

Let us be given the following TIFIMs, in accordance with the problem: The initial IM C , which consisting the time for passing the road from settlement k_i to settlement l_j at the current moment t_f has the form:

$$C[K, L, t_f] = \begin{matrix} & t_f & \dots & l_n & & R \\ \begin{matrix} k_1 \\ \vdots \\ k_m \\ Q \end{matrix} & \dots & \langle \mu_{k_1, l_n, t_f}, \nu_{k_1, l_n, t_f} \rangle & \dots & \langle \mu_{k_m, R, t_f}, \nu_{k_m, R, t_f} \rangle \\ & \ddots & \vdots & & \vdots \\ & \dots & \langle \mu_{k_m, l_n, t_f}, \nu_{k_m, l_n, t_f} \rangle & \dots & \langle \mu_{Q, R, t_f}, \nu_{Q, R, t_f} \rangle \end{matrix},$$

where t_f is a current moment, $K = \{k_1, k_2, \dots, k_m, Q\}$, $L = \{l_1, l_2, \dots, l_n, R\}$ and for $1 \leq i \leq m, 1 \leq j \leq n : \{c_{k_i, l_j, t_f}, c_{k_i, R, t_f}, c_{Q, l_j, t_f}\}$ are TIFPs. The time $c_{k_i, k_i, t_f} = \langle \perp, \perp \rangle$ (for $1 \leq i \leq m$). We can see, that $|K| = m + 1$ and $|L| = n + 1$;

$$X[K^0, L^0, t_f, \{x_{k_i, l_j, t_f}\}],$$

where $K^0 = \{k_1, k_2, \dots, k_m\}$, $L^0 = \{l_1, l_2, \dots, l_n\}$ and for $1 \leq i \leq m, 1 \leq j \leq n$:

$$x_{k_i, l_j, t_f} = \begin{cases} \langle 1, 0 \rangle, & \text{if the salesman travels from } k_i\text{-th} \\ & \text{settlement to } l_j\text{-th settlement at } t_f (k_i \neq l_j) \\ \langle \perp, \perp \rangle, & \text{if } k_i = l_j \\ \langle 0, 1 \rangle, & \text{otherwise.} \end{cases}$$

The auxiliary matrices in the algorithm have the forms:

- 1) $S(t_f) = [K, L, \{s_{k_i, k_j, t_f}\}]$, such that $S = C$.
- 2) $D[K^0, L^0, t_f]$ where for $1 \leq i \leq m, 1 \leq j \leq n$:

$$d_{k_i, l_j, t_f} \in \{1, 2\},$$

if the element s_{k_i, l_j, t_f} of S is crossed out with 1 or 2 lines respectively;

$$3) RC[K^0, e_0, t_f] = \begin{matrix} & t_f & e_0 \\ \begin{matrix} k_1 \\ \vdots \\ k_m \end{matrix} & \begin{matrix} rc_{k_1, e_0, t_f} \\ \vdots \\ rc_{k_m, e_0, t_f} \end{matrix} \end{matrix},$$

where $K^0 = \{k_1, k_2, \dots, k_m\}$ and for $1 \leq i \leq m$: rc_{k_i, e_0, t_f} is equal to 0 or 1, depending on whether the k_i -th row of K^0

of the matrix S is crossed out;

$$4) CC[r_0, L^0, t_f] = \begin{matrix} t_f & l_1 & \dots & l_j & \dots & l_n \\ r_0 & cc_{r_0, l_1, t_f} & \dots & cc_{r_0, l_j, t_f} & \dots & cc_{r_0, l_n, t_f} \end{matrix},$$

where $L^0 = \{l_1, l_2, \dots, l_n\}$ and for $1 \leq j \leq m$: cc_{k_i, l_j, t_f} is equal to 0 or 1, depending on whether the l_j -th row of L^0 of the matrix S is crossed out. Let us $rc_{k_i, e_0, t_f} = cc_{r_0, l_j, t_f} = 0$. In the beginning of the algorithm, the initial IM $C[K, L, t_f]$, which consists of the time evaluations $c_{k_i, l_j, t_f} = \langle \mu_{k_i, c_j, t_f}, \nu_{k_i, c_j, t_f} \rangle$ for passing the road from settlement k_i to settlement l_j at a current moment t_f are constructed by application of similar expert approach, described in [3], [8], [19]. The experts $E = \{d_1, \dots, d_s, \dots, d_D\}$ (for $s = 1, \dots, D$) are evaluated the time for passing the road from settlement k_i to settlement l_j at a current moment t_f as a TIFP $\{c_{k_i, l_j, d_s, t_f}\}$, interpreted as the degrees of perception (a positive time evaluation that the expert is more likely to accept for passing the road of the d_s -th expert for the k_i -th settlement to the l_j -th settlement divided by the *max-min* evaluation of time) and non-perception (a negative time evaluation for passing the road of the d_s -th expert from the k_i -th settlement to the l_j -th settlement divided by the *max-min* evaluation of time) of time evaluation for passing the road of the d_s -th expert from the k_i -th settlement to the l_j -th settlement at a current time t_f .

The hesitation degree $\mu_{k_i, l_j, d_s, t_f} = 1 - \mu_{k_i, l_j, d_s, t_f} - \nu_{k_i, l_j, d_s, t_f}$ corresponds to the uncertain evaluation of the time for passing the road of the d_s -th expert from the k_i -th settlement to the l_j -th settlement at a current time moment t_f . If the optimistic (pessimistic) scenario is obtained, then the maximum (minimum) degree of membership μ_{k_i, l_j, d_s, t_f} , proposed by the experts, will be the time evaluation for passing the road from the k_i -th settlement to the l_j -th settlement at a current time t_f . The degree of non-membership of the time evaluation for passing the road from the k_i -th settlement to the l_j -th settlement at a current time t_f is calculated as 1-(minimum) maximum degree of membership, proposed by the experts.

The index matrix interpretation of the described process for creating of the IM C is as follows:

A TIFIMs $C^s(t_f)[K, L, t_f, \{c_{s k_i, l_j, t_f} = \langle \mu_{k_i, l_j, t_f}, \nu_{k_i, l_j, t_f} \rangle\}]$ is created with the dimensions $K = \{k_1, k_2, \dots, k_m\}$, $L = \{l_1, l_2, \dots, l_n\}$ at a particular moment t_f for each expert $d_s (s = 1, \dots, D)$, where $K = \{k_1, k_2, \dots, k_m\}$, $L = \{l_1, l_2, \dots, l_n\}$ and the element $\{c_{s k_i, c_j, t_f}^s\}$ is the time evaluation for passing the road of the d_s -th expert from the k_i -th settlement to the c_j -th settlement at a current time t_f .

Let the ordinal coefficient $r_s(t_f)$ of each expert ($s \in D$) is defined by an TIFP $\langle \delta_s(t_f), \varepsilon_s(t_f) \rangle$, which elements can be interpreted respectively as his degree of competence and of incompetence at a current time t_f . Then the IM

$$C(t_f)[K, L, t_f, \{c_{k_i, l_j, t_f}\}] = r_1 C^1(t_f) \oplus_{(\#_q)} r_d C^d(t_f) \dots \oplus_{(\#_q)} r_D C^D(t_f),$$

$\forall k_i \in K, \forall l_j \in L, \forall s \in D, 1 \leq q \leq 3$ is constructed and it contains the final time evaluation for passing the road from the k_i -th settlement to the l_j -th settlement at a current time t_f .

If we use $\#_1^* = \langle \min, \max \rangle$, then the decision maker accepts super pessimistic scenario, with $\#_2^* = \langle \text{average}, \text{average} \rangle$ the decision maker assumes averaging scenario and with $\#_3^* =$

(\max, \min) he proposes super optimistic scenario for the time evaluation between the settlements.

The steps of TIFHA-TSP approach are as follows:

Step 1. This step checks the problem balance requirement according to [6]. For this aim, the algorithm compares the number of rows with the number of columns in C .

Step 1.1. If the number of rows is greater than the number of columns, then a dummy column $l_{n+1} \in \mathcal{L}$ is entered in the matrix C , in which all time evaluations $c_{k_i, l_{n+1}}$ ($i = 1, \dots, m$) are equal to $\langle 1, 0 \rangle$; otherwise, go to the *Step 1.2*. For this purpose, the following operations are executed:

– we define the IM $C_1[K/\{Q\}, l_{n+1}, t_f], \{c_{1, k_i, l_{n+1}, t_f}\}$, whose elements are equal to $\langle 1, 0 \rangle$;

– the new cost matrix is obtained by:

$C := C \oplus_{(\max, \min)} C_1; s_{k_i, l_j, t_f} = c_{k_i, l_j, t_f}, \forall k_i \in K, \forall l_j \in L$, Go to *Step 2*.

Step 1.2. If the number of columns is greater than the number of rows, then a dummy row $k_{m+1} \in \mathcal{K}$ is entered in the time matrix, in which all time evaluations are equal to $\langle 1, 0 \rangle$. Similar operations to those in *Step 1.1*. are performed. Let us create IM $S = [K, L, \{s_{k_i, l_j}\}]$ such that $S = C$.

Step 2. In each row k_i of K of S , the smallest element is found among the elements s_{k_i, l_j, t_f} ($j = 1, \dots, n$) and it is subtracted from all elements s_{k_i, l_j, t_f} for $j = 1, 2, \dots, n$.

Go to *Step 3*.

Step 2.1. For each row k_i of K of S , the smallest element is found and is recorded as the value of the element s_{k_i, R, t_f} :

for $i = 1$ to m , for $j = 1$ to n

$\{AGIndex_{\min_R, (\mathcal{L})}(pr_{k_i, L, t_f} S) = \langle k_i, l_{v_j}, t_f \rangle\}$;

If the minimum elements are more than one, then one of them is chosen arbitrary.

We create S_1 and S_2 : $S_1[k_i, l_{v_j}, t_f] = pr_{k_i, l_{v_j}, t_f} S$;

$S_2 = \left[\perp; \frac{R}{l_{v_j}}; \perp; \perp \right] S_1$; $S := S \oplus_{(\#_q)} S_2$, where $1 \leq q \leq 3$.

Step 2.2. The smallest element s_{k_i, l_{v_j}, t_f} is subtracted from the elements s_{k_i, l_j, t_f} ($j = 1, \dots, m$). Let us create IM $B = pr_{K, R, t_f} S$.

for $i = 1$ to m , for $j = 1$ to n

If $s_{k_i, l_j, t_f} \neq \perp$, then $\{IO_{-(\max, \min)}(\langle k_i, l_j, t_f, S \rangle, \langle k_i, R, t_f, B \rangle)\}$.

Step 3. For each index l_j of L of S , the smallest element is found among the elements s_{k_i, l_j, t_f} ($i = 1, \dots, m$) and it is subtracted from all elements s_{k_i, l_j, t_f} , for $i = 1, 2, \dots, m$ at a particular moment t_f . Go to *Step 4*. Similar operations to those in *Step 2*. are executed. They are presented in [16].

Step 4. At this step are crossed out all elements s_{k_i, l_j, t_f} for $\langle k_i, l_j, t_f \rangle \in \{Index_{(\max v), k_i/l_j}(S)\}$ or equal to $\langle 0, 1 \rangle$ in S with the minimum possible number of lines (horizontal, vertical or both). If the number of these lines is m , go to *Step 6*. If the number of lines is less than m , go to *Step 7*.

This step introduces IM $D[K^0, L^0, t_f]$, which has the same structure as the IM X . We use to mark whether an element in S is crossed out with a line.

If $d_{k_i, l_j, t_f} = 1$, then s_{k_i, l_j, t_f} is covered with one line;

if $d_{k_i, l_j, t_f} = 2$, then s_{k_i, l_j, t_f} is covered with two lines.

The IMs $CC[r_0, L^0]$ and $RC[K^0, e_0]$ reflect whether the element is covered by a line in a row or column in the S matrix.

for $i = 1$ to m , for $j = 1$ to n

If $s_{k_i, l_j, t_f} = \langle 0, 1 \rangle$ (or $\langle k_i, l_j, t_f \rangle \in Index_{(\max v), k_i/l_j}(S)$) and $d_{k_i, l_j, t_f} = 0$, then $\{rc_{k_i, e_0, t_f} = 1$; for $i = 1$ to m $d_{k_i, l_j} = 1$; $S_{(k_i, \perp, t_f)}\}$.

If $s_{k_i, l_j, t_f} = \langle 0, 1 \rangle$ (or $\langle k_i, l_j, t_f \rangle \in Index_{(\max v), k_i}(S)$) and $d_{k_i, l_j, t_f} = 1$, then $\{d_{k_i, l_j, t_f} = 2$; $cc_{r_0, l_j, t_f} = 1$;

for $j = 1$ to n $d_{k_i, l_j, t_f} = 1$; $S_{(\perp, l_j, t_f)}\}$.

Then we count the covered rows and columns in CC and RC :

$Index_{(1)}(RC) = \{\langle k_{u_1}, e_0, t_f \rangle, \dots, \langle k_{u_i}, e_0, t_f \rangle, \dots, \langle k_{u_x}, e_0, t_f \rangle\}$;

$Index_{(1)}(CC) = \{\langle r_0, l_{v_1}, t_f \rangle, \dots, \langle r_0, l_{v_j}, t_f \rangle, \dots, \langle r_0, l_{v_y}, t_f \rangle\}$.

If $\text{count}(Index_{(1)}(RC)) + \text{count}(Index_{(1)}(CC)) = m$, then go to *Step 6*, otherwise to *Step 5*.

Step 5. We find the smallest element in the IM S that it is not crossed by the lines in *Step 4*, and subtract it from any uncovered element of S , and we add it to each element, which is covered by two lines. We return to *Step 4*.

The operation $AGIndex_{\min_R, (\mathcal{L})}(S) = \langle k_x, l_y, t_f \rangle$ finds the smallest element index of the IM S . The operation $IO_{-(\max, \min)}(\langle S \rangle, \langle k_x, l_y, t_f S \rangle)$ subtract it from any uncovered element of S . Then we add it to each element of S , which is crossed out by two lines:

for $i = 1$ to m , for $j = 1$ to n

{if $d_{k_i, l_j, t_f} = 2$, then $S_1 = pr_{k_x, l_y, t_f} C$;

$S_2 = pr_{k_i, l_j, t_f} C \oplus_{(\#_q)} \left[\frac{k_i}{k_x}, \frac{l_j}{l_y}, t_f \right] S_1$; $S := S \oplus_{(\#_q)} S_2$;

if $d_{k_i, l_j, t_f} = 1$ or $d_{k_i, l_j, t_f} = 2$ then $S := S \oplus_{(\#_q)} pr_{k_i, l_j, t_f} C$;

Go to *Step 4*.

Step 6. Here we search each row until we find a row-wise exactly single element, which is a TIFFalseP. We mark this pair to make the assignment, then cross out all elements lying in the respective column. If there lie more than one unmarked TIFFalseP in any column or row, then choose one of them arbitrary and cross out the remaining elements in its row or column. We repeat until no unmarked elements are left in the reduced IM. Then each row and column of S has exactly one marked TIFFalseP. If the solution does not satisfy the route conditions, adjustments need to be made to the assignments with minimum increase to the total cost [14]. The optimal solution $X_{opt}[K^0, L^0, \{x_{k_i, l_j, t_f}\}]$ is found where the elements $\langle 1, 0 \rangle$ in X correspond to the marked elements in S .

{for $i = 1$ to m , for $j = 1$ to n

if $(\langle k_i, l_j, t_f \rangle \in Index_{(\max v), k_i/l_j}(S))$ and $d_{k_i, l_j, t_f} \neq 3$, then $x_{k_i, l_j, t_f} = \langle 1, 0 \rangle$;

for $i = 1$ to m $d_{k_i, l_j, t_f} = 3$

for $j = 1$ to n $d_{k_i, l_j, t_f} = 3$ };

The optimal time evaluation for X_{opt} at a particular moment t_f is:

$$AGIO_{\oplus(\#_q)} \left(C(\{Q\}, \{R\}) \otimes_{(\#_q)} X_{opt} \right),$$

$\forall k_i \in K, \forall l_j \in L, 1 \leq q \leq 3$ in a pessimistic scenario $q = 1$. If we want an optimistic result, we can use 3 in place of 1.

Step 7. The old rank coefficients of the experts, involved in the evaluation process are changed by [8]:

$$= \begin{cases} \left\langle \frac{\delta(t_f)\lambda+1}{\lambda+1}, \frac{\varepsilon(t_f)\lambda}{\lambda+1} \right\rangle, & \text{if the expert has assessed correctly} \\ \left\langle \frac{\delta(t_f)\gamma}{\lambda+1}, \frac{\varepsilon(t_f)\lambda}{\lambda+1} \right\rangle, & \text{if the expert had not given any estimation} \\ \left\langle \frac{\delta(t_f)\lambda}{\lambda+1}, \frac{\varepsilon(t_f)\lambda+1}{\lambda+1} \right\rangle, & \text{if the expert has assessed incorrectly} \end{cases} \quad (3)$$

The complexity of the HA as used here is comparable with that of the standard Hungarian method [7], [13]. To explore the effect of TIFHA-TSP on various input data, we are currently developing a software tool that implements the algorithm. It reads a file containing the matrix of time evaluations and performs the steps stated above. It is written in C++ and uses the *IndexMatrix* template class [4] with TIFPs. Several operations had to be adjusted, most significantly the IFP and TIFP subtraction operation, which will now return a false pair $\langle 0,1 \rangle$ of the two operands are equal, and subtracting a false pair from another (T)IFP will result in no change.

B. A Real Case Study of TIFHA-TSP

In this section, we demonstrate TIFHA-TSP with a real case study in a shipping company for a day of the week in Bulgaria. The carrier needs to visit 4 settlements and wants to make a cyclical route passing all of them. The evaluation TIFPs are given by experts with ratings $\{\langle 0.9,0.05 \rangle, \langle 0.8,0.05 \rangle, \langle 0.95,0.05 \rangle\}$ defined on the basis of their participation in 10 ($s = 1, \dots, D$) past procedures respectively in the days before t_f . The initial evaluation time IM $C[K, L, t_f]$ incorporating the ratings of the experts is:

$$\begin{matrix} \left. \begin{matrix} t_f & l_1 & l_2 & l_3 & l_4 & R \\ k_1 & \langle \perp, \perp \rangle & \langle 0.72, 0.09 \rangle & \langle 0.55, 0.2 \rangle & \langle 0.45, 0.45 \rangle & \langle \perp, \perp \rangle \\ k_2 & \langle 0.57, 0.23 \rangle & \langle \perp, \perp \rangle & \langle 0.7, 0.15 \rangle & \langle 0.61, 0.1 \rangle & \langle \perp, \perp \rangle \\ k_3 & \langle 0.67, 0.15 \rangle & \langle 0.4, 0.28 \rangle & \langle \perp, \perp \rangle & \langle 0.7, 0.05 \rangle & \langle \perp, \perp \rangle \\ k_4 & \langle 0.83, 0.05 \rangle & \langle 0.75, 0.1 \rangle & \langle 0.65, 0.13 \rangle & \langle \perp, \perp \rangle & \langle \perp, \perp \rangle \\ Q & \langle \perp, \perp \rangle & \langle \perp, \perp \rangle & \langle \perp, \perp \rangle & \langle \perp, \perp \rangle & \langle \perp, \perp \rangle \end{matrix} \right\} \end{matrix}$$

After application of the software utility, described in the section (III), we get the following results:

Step 1. $m = n$, therefore the problem is balanced.

Step 2. - 3. In each row k_i of K of S , the smallest element is found among the elements s_{k_i, l_j, t_f} ($j = 1, \dots, n$) and it is subtracted from all elements s_{k_i, l_j, t_f} for $j = 1, 2, \dots, n$. For each index l_j of L of S , the smallest element is found among the elements s_{k_i, l_j, t_f} ($i = 1, \dots, m$) and it is subtracted from all elements s_{k_i, l_j, t_f} , for $i = 1, 2, \dots, m$ at a particular moment t_f . The form of the IM $C(t_f)$ is

$$\left. \begin{matrix} \begin{matrix} l_1 & l_2 & l_3 & l_4 & R \\ k_1 & \langle \perp, \perp \rangle & \langle 0.27, 0.54 \rangle & \langle 0.1, 0.65 \rangle & \langle 0, 1 \rangle & \langle 0.45, 0.45 \rangle \\ k_2 & \langle 0, 1 \rangle & \langle \perp, \perp \rangle & \langle 0.13, 0.38 \rangle & \langle 0.04, 0.33 \rangle & \langle 0.57, 0.23 \rangle \\ k_3 & \langle 0.27, 0.43 \rangle & \langle 0, 1 \rangle & \langle \perp, \perp \rangle & \langle 0.3, 0.33 \rangle & \langle 0.4, 0.28 \rangle \\ k_4 & \langle 0.18, 0.18 \rangle & \langle 0.1, 0.23 \rangle & \langle 0, 1 \rangle & \langle \perp, \perp \rangle & \langle 0.65, 0.13 \rangle \\ Q & \langle 0, 1 \rangle & \langle 0, 1 \rangle & \langle 0, 1 \rangle & \langle 0, 1 \rangle & \langle \perp, \perp \rangle \end{matrix} \end{matrix} \right\}$$

Step 4. At this step are crossed out all elements s_{k_i, l_j, t_f} equal to $\langle 0,1 \rangle$ in S with the minimum possible number of lines. Since each row and each column contain exactly one false pair, we will cross out all four rows. The test for the optimality is satisfied (the number of lines = $m = n = 4$). Step 5 does not apply, because there are no uncrossed elements.

The next steps will give us the final results.

In **Step 6**, we find that the optimal route is: $A \rightarrow D \rightarrow C \rightarrow B \rightarrow A$ and in an optimistic scenario it will take the optimal time of travelling with degree of membership 0.65 and degree of non-membership 0.13, forming the IFP $\langle 0.65, 0.13 \rangle$. In a pessimistic scenario, the optimal time IFP is $\langle 0.4, 0.45 \rangle$.

Step 7. The new ranking coefficients of the experts are equal respectively to $\{\langle 0.91, 0.05 \rangle, \langle 0.81, 0.05 \rangle, \langle 0.95, 0.05 \rangle\}$, because their evaluations are intuitionistic fuzzy correct.

IV. CONCLUSION

In the study, we defined TIFHA-TSP, a temporal intuitionistic fuzzy extension of the Hungarian method for solving the TSP using the IF logic and IMs concepts. The developed software, which implements this TIFHA-TSP approach, is applied to a real case in a shipping company at a certain time. In the future, the research will continue with the development of TIFHA-TSP so that it can find the optimal solution of the IFTSP, where the initial data have saved in extended TIFIMs [9] and also with a software for its implementation.

REFERENCES

- [1] A. Mucherino, S. Fidanova, M. Ganzha, "Ant colony optimization with environment changes: An application to GPS surveying," *Proceedings of the 2015 FedCSIS*, 2015, pp. 495 - 500.
- [2] A. Sudha, G. Angel, M. Priyanka, S. Jennifer, "An Intuitionistic Fuzzy Approach for Solving Generalized Trapezoidal Travelling Salesman Problem," *International Journal of Mathematics Trends and Technology*, vol. 29 (1), 2016, pp. 9-12.
- [3] D. Mavrov, V. Atanassova, V. Bureva, O. Roeva, P. Vassilev, R. Tsvetkov, D. Zoteva, E. Sotirova, K. Atanassov, A. Alexandrov, H. Tsakov, "Application of Game Method for Modelling and Temporal Intuitionistic Fuzzy Pairs to the Forest Fire Spread in the Presence of Strong Wind," *Mathematics*, vol. 10, 2022, 1pp. 1280. <https://doi.org/10.3390/mat10081280>
- [4] D. Mavrov, "An Application for Performing Operations on Two-Dimensional Index Matrices," *Annual of "Informatics" Section, Union of Scientists in Bulgaria*, vol. 10, 2019 / 2020, pp. 66-80.
- [5] E. Szmidt, J. Kacprzyk, "Amount of information and its reliability in the ranking of Atanassov's intuitionistic fuzzy alternatives," in: *Rakus-Andersson, E., Yager, R., Ichalkaranje, N., Jain, L.C. (eds.), Recent Advances in Decision Making*, SCI, Springer, vol. 222, 2009, pp. 7-19.
- [6] H. Kuhn, *The Travelling salesman problem*, Proc. Sixth Symposium in Applied Mathematics of the American Mathematical Society, McGraw-Hill, New York; 1955
- [7] J. Wong, "A new implementation of an algorithm for the optimal assignment problem: An improved version of Munkres' algorithm," *BIT*, vol. 19, 1979, pp. 418-424.
- [8] K. Atanassov, *On Intuitionistic Fuzzy Sets Theory*, STUDEFUZZ. Springer, Heidelberg, vol. 283; 2012. DOI:10.1007/978-3-642-29127-2.
- [9] K. Atanassov, "Index Matrices: Towards an Augmented Matrix Calculus," *Studies in Computational Intelligence*, Springer, vol. 573, 2014.
- [10] K. Prabaharan, K. Ganesan, "Fuzzy Hungarian method for solving intuitionistic fuzzy travelling salesman problem," *Journal of Physics: Conf. Series*, vol. 1000, 2018, pp. 2-13.
- [11] L. Biggs, K. Lloyd, R. Wilson, *Graph Theory 1736-1936*, Clarendon Press, Oxford; 1986.
- [12] L. Zadeh, "Fuzzy Sets," *Information and Control*, vol. 8 (3), 1965, pp. 338-353.
- [13] S. Akshitha, K. S. Ananda Kumar, M. Nethrithameda, R. Sowmva, and R. Suman Pawar, "Implementation of hungarian algorithm to obtain optimal solution for travelling salesman problem," 2018, pp. 2470-2474.
- [14] S. Dhanasekar, S. Hariharan, P. Sekar, "Classical travelling salesman problem based approach to solve fuzzy TSP using Yager's Ranking," *Int. Journal of Computer Applications*, vol. 74 (13), 2013, pp. 1-4.
- [15] S. Dhanasekar, V. Parthiban, D. Gururaj, "Improved Hungarian method to solve fuzzy assignment problem and fuzzy TSP," *Advances in Mathematics: Scientific Journal*, vol. 9 (11), 2021, pp. 9417-9427.
- [16] V. Traneva, S. Tranev, V. Atanassova, "An Intuitionistic Fuzzy Approach to the Hungarian Algorithm," in: *G. Nikolov et al. (Eds.): NMA 2018*, LNCS 11189, Springer Nature Switzerland, AG, 2019, pp. 1-9.
- [17] V. Traneva, S. Tranev, M. Stoenchev, K. Atanassov, " Scaled aggregation operations over two- and three-dimensional index matrices," *Soft computing*, vol. 22, 2019, pp. 5115-5120.
- [18] V. Traneva, S. Tranev, *Index Matrices as a Tool for Managerial Decision Making*, Publ. House of the USB; 2017 (in Bulgarian).
- [19] V. Traneva, S. Tranev, "An Intuitionistic Fuzzy Approach to the Travelling Salesman Problem," In: *Lirkov, I., Margenov, S. (eds) Large-Scale Scientific Computing, LSSC 2019*, Lecture Notes in Computer Science, vol. 11958. Springer, Cham, 2021, pp. 530-539.

Software Implementation of the Optimal Temporal Intuitionistic Fuzzy Algorithm for Franchisee Selection

Velichka Traneva

Prof. Asen Zlatarov University
 1 Prof. Yakimov Blvd, Burgas 8000, Bulgaria
 Email: veleka13@gmail.com

Deyan Mavrov

Prof. Asen Zlatarov University
 1 Prof. Yakimov Blvd, Burgas 8000, Bulgaria
 Email: dg@mavrov.eu

Stoyan Tranev

Prof. Asen Zlatarov University
 1 Prof. Yakimov Blvd, Burgas 8000, Bulgaria
 Email: tranev@abv.bg

Abstract—The selection of the most suitable franchisee applicant in an uncertain environment in a particular moment of time is a key decision for a franchisor and the success of a franchising business. In this work, for the first time, we describe a problem for choosing the optimal candidate for the franchise chain and algorithm for a solution in terms of temporal intuitionistic fuzzy pairs and index matrices as a means for data analysis in uncertain conditions over time. We also use our software utility to demonstrate the proposed algorithm and to apply the decision support approach to a franchisee selection for the largest fast food restaurant chain in Bulgaria.

I. INTRODUCTION

Franchising is an effective business strategy for entering new markets. The franchisor grants the right to its franchisees to use the brand, the business concept, and the products or services within a specific time frame [1]. The concept of fuzzy [10] and intuitionistic fuzzy logic [6], provides such a tool for creating an optimal algorithm for choosing a franchisee in conditions of ambiguity. The studies [12], [13] present fuzzy franchisee selection models using an Analytic Hierarchy Process (AHP) and neural networks.

In [16], we presented an optimal interval-valued intuitionistic fuzzy multicriteria decision-making problem in outsourcing and a software utility for its solution. We have also introduced in [19] an intuitionistic fuzzy approach (IFIMFr) to select the most suitable candidates for franchising in a patisserie using the theory of index matrices (IMs, [5]). The aim of the paper is to expand the IFIMFr approach so that it can be applied to temporal intuitionistic fuzzy data [3]. The work uses our custom programs to implement the proposed algorithm and to apply it to the largest fast food restaurant chain in Bulgaria. The remainder of our study includes 4 sections: Section II describes some definitions and properties of temporal intuitionistic fuzzy IMs and pairs. Section III describes a problem for choosing the optimal franchise candidate and algorithm

for solution in terms of temporal IFPs an IMs as a means for uncertain data analysis over time and the basic characteristics of our software utility. Section IV sets out the conclusions and aspects for future research.

II. DEFINITION AND PROPERTIES OF TEMPORAL INTUITIONISTIC FUZZY IMs AND PAIRS

Let us briefly give the definitions of temporal intuitionistic fuzzy IMs and TIFPs and some of their properties [3].

2.1. Temporal Intuitionistic Fuzzy Pair (TIFP)

Let $T = \{t_1, \dots, t_g, \dots, t_f\}$ be a fixed time-scale. A TIFP is in the form of $\langle \mu(t), \nu(t) \rangle$, where $\mu(t)$ and $\nu(t)$ are interpreted as degrees of membership and non-membership, $\mu, \nu : T \rightarrow [0, 1]$ and $\mu(t) + \nu(t) \leq 1$ for $t \in T$. Let us have two TIFPs $x = \langle \mu(t), \nu(t) \rangle$ and $y = \langle \rho(t), \sigma(t) \rangle$. Then, we recall some basic operations [3] with two TIFPs.

$$\begin{aligned} x(t) \wedge_1 y(t) &= \langle \min(\mu(t), \rho(t)), \max(\nu(t), \sigma(t)) \rangle \\ x(t) \vee_1 y(t) &= \langle \max(\mu(t), \rho(t)), \min(\nu(t), \sigma(t)) \rangle; \\ x(t) \wedge_2 y(t) &= x(t) + y(t) = \langle \mu(t) + \rho(t) - \mu(t) \cdot \rho(t), \nu(t) \cdot \sigma(t) \rangle \end{aligned} \quad (1)$$

Let $R_{\langle a(t), b(t) \rangle} = 0.5 \cdot (2 - a(t) - b(t)) \cdot (1 - a(t))$ following [4]. Then, as per [6], [20]:

$$x(t) \geq_R y(t) \text{ iff } R_{\langle a(t), b(t) \rangle} \leq R_{\langle c(t), d(t) \rangle}. \quad (2)$$

2.2. Three-Dimensional Temporal Intuitionistic Fuzzy Index Matrices (3-D TIFIM)

A 3-D TIFIM [5] $A(T) = [K, L, T, \{\langle \mu_{k_i, l_j, t_g}, \nu_{k_i, l_j, t_g} \rangle\}]$

$$= \begin{array}{c|ccc} t_g \in T & l_1 & \dots & l_n \\ \hline k_1 & \langle \mu_{k_1, l_1, t_g}, \nu_{k_1, l_1, t_g} \rangle & \dots & \langle \mu_{k_1, l_n, t_g}, \nu_{k_1, l_n, t_g} \rangle \\ \vdots & \vdots & \dots & \vdots \\ k_m & \langle \mu_{k_m, l_1, t_g}, \nu_{k_m, l_1, t_g} \rangle & \dots & \langle \mu_{k_m, l_n, t_g}, \nu_{k_m, l_n, t_g} \rangle \end{array}, \quad (3)$$

where \mathcal{S} be a fixed set of indices, $(K, L, T \subset \mathcal{S})$, and its elements $\langle \mu_{k_i, l_j, t_g}, \nu_{k_i, l_j, t_g} \rangle$ are TIFPs. T is a fixed temporal scale and its elements $t_g (g = 1, \dots, g, \dots, f)$ are time moments.

In [5], [17], [18], operations with 3-D TIFIMs, analogous to those with the classical matrices were introduced, but there are also specific ones such as projection, substitution, aggregation operations, internal subtraction of IMs' components, term-wise multiplication and subtraction. Let us recollect some operations with an application in temporal IFIMFr.

Aggregation operation by one dimension [17]: Let us have

Work is supported by the Asen Zlatarov University through project Ref. No. NIX-449/2021 "Index matrices as a tool for knowledge extraction".

two TIFPs $x = \langle a, b \rangle$ and $y = \langle c, d \rangle$ and $(1 \leq q \leq 3)$. An aggregation operation by one dimension is

$$\alpha_{K, \#_q}(A(T), k_0) = \begin{array}{c|ccc} t_g & l_1 & \dots & l_n \\ \hline k_0 & \#_q^m \langle \mu_{k_i, l_1, t_g}, \nu_{k_i, l_1, t_g} \rangle & \dots & \#_q^m \langle \mu_{k_i, l_n, t_g}, \nu_{k_i, l_n, t_g} \rangle \end{array} \quad (4)$$

If we use $\#_1^* = \langle \min(a(t), c(t)), \max(b(t), d(t)) \rangle$ we perform a super pessimistic aggregation operation, with $\#_2^* = \langle \text{average}(a(t), c(t)), \text{average}(b(t), d(t)) \rangle$ we have an averaging aggregation operation, and with $\#_3^* = \langle \max(a(t), c(t)), \min(b(t), d(t)) \rangle$ we perform a super optimistic aggregation operation.

Projection: Let $W \subseteq K$, $V \subseteq L$ and $U \subseteq H$. Then, $pr_{W, V, U}A(T) = [W, V, U, \{\langle R_{pr, q_s, e_d}, S_{pr, q_s, e_d} \rangle\}]$, where for each $k_i \in W, l_j \in V$ and $t_g \in U$, $\langle R_{pr, q_s, e_d}, S_{pr, q_s, e_d} \rangle = \langle \mu_{k_i, l_j, t_g}, \nu_{k_i, l_j, t_g} \rangle$.

A Level Operator for Decreasing the Number of Elements of TIFIM: Let $\langle \alpha(t), \beta(t) \rangle$ is an TIFP, then according to [9] $N_{\alpha(t), \beta(t)}^>(A(T)) = [K, L, T, \{\langle \rho_{k_i, l_j, t_g}, \sigma_{k_i, l_j, t_g} \rangle\}]$, where

$$= \begin{cases} \langle \rho_{k_i, l_j, t_g}, \sigma_{k_i, l_j, t_g} \rangle & \text{if } \langle \rho_{k_i, l_j, t_g}, \sigma_{k_i, l_j, t_g} \rangle > \langle \alpha(t), \beta(t) \rangle \\ (0, 1) & \text{otherwise} \end{cases} \quad (5)$$

III. AN OPTIMAL TEMPORAL INTUITIONISTIC FUZZY ALGORITHM FOR SELECTION OF THE MOST ELIGIBLE FRANCHISEE (OTIFAFr)

This section proposes an OTIFAFr, used the concepts of IMs and TIFPs. Let us formulate the optimal problem as follows:

A large franchise has decided to turn to experts to select the best franchisee candidate for expanding its brand. The franchise candidates for studied brand v_e need to be evaluated by the experts. The experts assess the IF priorities pk_{c_j, v_e, t_f} of the criteria c_j in the evaluation system of the franchise chain v_e at a particular moment t_f . The IF ratings of the experts are defined on the basis of their participation in previous franchise evaluation procedures and given to the experts at a time t_f . All candidates for a franchisee have been evaluated by the experts at a particular moment t_f and their evaluations ev_{k_i, c_j, d_s, t_f} at a current moment t_f are temporal intuitionistic fuzzy data. The estimations of the same applicants from previous evaluation procedures are given as elements of TIFIM at time points $t_1, \dots, t_g, \dots, t_{f-1}$. The aim of the problem is to select the most eligible franchisee for this brand.

A. An Optimal Temporal Intuitionistic Fuzzy Algorithm for Assignment of a Franchisee

The procedure of OTIFAFr includes the following steps:

Step 1. The team of experts needs to evaluate the candidates for the brand v_e according to the approved criteria in the company at a particular moment t_f . The estimations of the $d_s (1 \leq s \leq D)$ expert are described by the TIFP $ev_{k_i, c_j, d_s, t_f} = \langle \mu_{k_i, c_j, d_s, t_f}, \nu_{k_i, c_j, d_s, t_f} \rangle$ by criterion $c_j (1 \leq j \leq n)$ for the the k_i -th $(1 \leq i \leq m)$ candidate at a particular moment t_f . Expert assessments are uncertain due to galloping inflation and the existing pandemic. The data values are transformed into TIFPs as demonstrated in [3], [20]. The TIFP $\{ev_{k_i, c_j, d_s, t_f}\}$ presents the degrees of perception (the positive evaluation of the d_s -th expert for the k_i -th candidate by the c_j -th criterion divided by the (maximum-minimum) evaluation) and non-perception (the

negative evaluation of the d_s -th expert for the k_i -th candidate by the c_j -th criterion divided by the (maximum-minimum) evaluation) of the d_s -th expert for the k_i -th candidate by the c_j -th criterion at a particular moment t_f . The hesitation degree $\mu_{k_i, c_j, d_s, t_f} = 1 - \mu_{k_i, c_j, d_s, t_f} - \nu_{k_i, c_j, d_s, t_f}$ corresponds to the uncertain evaluation of the d_s -th expert for the k_i -th candidate by the c_j -th criterion at a particular moment t_f .

The experts have the opportunity to include assessments for the same candidates from the previous evaluation procedures at time points $t_1, \dots, t_g, \dots, t_{f-1}$. A TIFIM $EV_s[K, C, T, \{ev_{k_i, c_j, d_s, t_g}\}]$ is built with the dimensions $K = \{k_1, k_2, \dots, k_m\}$, $C = \{c_1, c_2, \dots, c_n\}$ and $T = \{t_1, t_2, \dots, t_f\}$ for each expert $d_s (s = 1, \dots, D)$:

$$EV_s[K, C, T, \{ev_{k_i, c_j, d_s, t_g}\}] = \begin{array}{c|ccc} t_g \in T & c_1 & \dots & c_n \\ \hline k_1 & \langle \mu_{k_1, c_1, d_s, t_g}, \nu_{k_1, c_1, d_s, t_g} \rangle & \dots & \langle \mu_{k_1, c_n, d_s, t_g}, \nu_{k_1, c_n, d_s, t_g} \rangle \\ \vdots & \vdots & \ddots & \vdots \\ k_m & \langle \mu_{k_m, c_1, d_s, t_g}, \nu_{k_m, c_1, d_s, t_g} \rangle & \dots & \langle \mu_{k_m, c_n, d_s, t_g}, \nu_{k_m, c_n, d_s, t_g} \rangle \end{array},$$

where $K = \{k_1, k_2, \dots, k_m\}$, $C = \{c_1, c_2, \dots, c_n\}$, $T = \{t_1, t_2, \dots, t_f\}$ and the element $\{ev_{k_i, c_j, d_s, t_g}\}$ is the estimate of the d_s -th expert for the k_i -th candidate by the c_j -th criterion at a particular moment t_g . Let us apply the α_T -th aggregation operation (4) to find the aggregated evaluation of the d_s -th expert $(s = 1, \dots, D)$ for the k_i -th candidate for the period T .

The result IM $\alpha_{EV_s(T), \#_q}$ has the form

$$\alpha_{EV_s(T), \#_q} = \begin{array}{c|ccc} d_s & \dots & c_n \\ \hline k_1 & \dots & \#_q^f \langle \mu_{k_1, c_n, d_s, t_g}, \nu_{k_1, c_n, d_s, t_g} \rangle_{g=1} & \dots \\ \vdots & \ddots & \vdots & \ddots \\ k_m & \dots & \#_q^f \langle \mu_{k_m, c_n, d_s, t_g}, \nu_{k_m, c_n, d_s, t_g} \rangle_{g=1} & \dots \end{array},$$

where $1 \leq q \leq 3$ depending on whether the pessimistic, averaging or optimistic scenarios are accepted in the process of decision making in the franchise chain. Then we create aggregated TIFIM $EV[K, C, E, \{ev_{k_i, c_j, d_s, t_f}\}]$ with the evaluations of all experts for all candidates by all criteria:

$$EV(t_f) = \alpha_{EV_1, \#_q}(T, d_1) \oplus_{(max, min)} \dots \oplus_{(max, min)} \alpha_{EV_D, \#_q}(T, d_D) \quad (6)$$

Then we go to Step 2.

Step 2. Let the present score coefficient $r_s(t_f)$ of each expert $(s \in E)$ is defined by an TIFP $\langle \delta_s(t_f), \varepsilon_s(t_f) \rangle$, which elements can be interpreted respectively as his degree of competence and of incompetence at a particular moment t_f . Then is created the IM $V(t_f)[K, C, E, \{v_{k_i, c_j, d_s, t_f}\}]$

$$= r_1 pr_{K, C, d_1, t_f} EV \oplus_{(max, min)} r_2 pr_{K, C, d_2, t_f} EV \dots \oplus_{(max, min)} r_D pr_{K, C, d_D, t_f} EV.$$

The IM $EV(t_f) := V(ev_{k_i, l_j, d_s, t_f} = v_{k_i, l_j, d_s, t_f}, \forall k_i \in K, \forall l_j \in L, \forall d_s \in E)$ contains the final score of each franchise candidate at a particular moment t_f .

The total assessment of the k_i -th candidate on the c_j -th criterion at a particular moment $t_f \notin E$ is calculated by an application of the α_E -th aggregation operation as follows

$$R(h_0) = \alpha_{E, \#_q}(EV(t_f), h_0) = \left\{ \begin{array}{c|ccc} c_j & h_0 & & \\ \hline k_1 & \#_q^D \langle \mu_{k_1, c_j, d_s, t_f}, \nu_{k_1, c_j, d_s, t_f} \rangle & & \\ \vdots & \vdots & & \\ k_m & \#_q^D \langle \mu_{k_m, c_j, d_s, t_f}, \nu_{k_m, c_j, d_s, t_f} \rangle & & \end{array} \right\}_{c_j \in C}, \quad (7)$$

where $(1 \leq q \leq 3)$.

If we use $\#_1^* = \langle \min, \max \rangle$, then we accept super pessimistic aggregation operation, with $\#_2^* = \langle \text{average}, \text{average} \rangle$ we assume averaging aggregation operation and with $\#_3^* = \langle \max, \min \rangle$ we accept super optimistic aggregation operation for the assessment of the applicant.

If the franchise chain has a requirement for the candidates, so that their total score is not less than a predetermined TIFP $\langle \alpha(t_f), \beta(t_f) \rangle$, then in this case it is necessary to apply the level operator (5) to TIFIM B to remove from the ranking candidates who do not meet this requirement. Go to *Step 3*.

Step 3. This step creates a TIFIM $PK(h_0)$ with the coefficients determining the importance of the evaluation criteria for the franchisor v_e at a particular moment t_f by :

$$PK(h_0)[C, v_e, h_0, \{pk_{c_j, v_e, h_0}\}] = \begin{array}{c|c} h_0 & v_e \\ \hline c_1 & pk_{c_1, v_e, h_0} \\ \vdots & \vdots \\ c_j & pk_{c_j, v_e, h_0} \\ \vdots & \vdots \\ c_n & pk_{c_n, v_e, h_0} \end{array},$$

Then we calculate the evaluation TIFIM

$$B(h_0)[K, v_e, h_0, \{b_{k_i, v_e, h_0}\}] = R(h_0) \odot_{(\circ, *)} PK(h_0),$$

containing the total estimates of the k_i -th candidate (for $1 \leq i \leq m$) at a particular moment t_f for the brand v_e , where $\langle \circ, * \rangle$ is an operation from (4). Go to *Step 4*.

Step 4. At this step we choose the most optimal franchisee for v_e by using the aggregation operation by $K - \alpha_{K, \#_q}(B(h_0), k_0)$ using pessimistic, average or optimistic scenarios

$$\alpha_{K, \#_q}(B(h_0), k_0) = \begin{array}{c|c} & v_e \\ \hline k_0 & \begin{array}{c} m \\ \#_q \\ \langle \mu_{k_i, v_e, h_0}, \nu_{k_i, v_e, h_0} \rangle \end{array} \end{array}, \quad (8)$$

where $k_0 \notin K, 1 \leq q \leq 3$. Go to *Step 5*.

Step 5. After finding the most effective franchisee, we will optimize the evaluation system for the next procedures using the intercriteria method (ICrA, [7], [8], [14]).

Let $\langle \alpha, \beta \rangle$ is an TIFP. The criteria C_k and C_l are in:

- $\langle \alpha(t_f), \beta(t_f) \rangle$ -positive consonance at a particular moment t_f , if $\mu_{C_k, C_l}(t_f) > \alpha(t_f)$ and $\nu_{C_k, C_l}(t_f) < \beta(t_f)$;
- $\langle \alpha(t_f), \beta(t_f) \rangle$ -negative consonance at a particular moment, if $\mu_{C_k, C_l}(t_f) < \beta(t_f)$ and $\nu_{C_k, C_l}(t_f) > \alpha(t_f)$;
- $\langle \alpha(t_f), \beta(t_f) \rangle$ -dissonance at a particular moment t_f , otherwise.

ICrA is applied over the matrix $R(h_0)$ to find the criteria, which are in a consonance. More complex criteria are reduced from the evaluation franchise system using the IM reduction operation over $R(h_0)$. Go to *Step 6*.

Step 6. This step obtains the new rank coefficients of the experts. Let the expert d_s ($s = 1, \dots, D$) has participated in γ_s evaluation procedures for the selection of a franchisee, on the basis of which his score $r_s(t_f) = \langle \delta_s(t_f), \varepsilon_s(t_f) \rangle$ is determined, then after his participation in the next procedure, his new score

$$\langle \delta_s^{(t_f+1)}, \varepsilon_s^{(t_f+1)} \rangle \text{ will be changed by [6]:} \begin{cases} \langle \frac{\delta(t_f)\gamma+1}{\gamma+1}, \frac{\varepsilon(t_f)\gamma}{\gamma+1} \rangle, & \text{if the expert has assessed correctly} \\ \langle \frac{\delta(t_f)\gamma}{\gamma+1}, \frac{\varepsilon(t_f)\gamma}{\gamma+1} \rangle, & \text{if the expert had not given any estimation} \\ \langle \frac{\delta(t_f)\gamma}{\gamma+1}, \frac{\varepsilon(t_f)\gamma+1}{\gamma+1} \rangle, & \text{if the expert has assessed incorrectly} \end{cases} \quad (9)$$

The complexity of OTIFAFr algorithm is $O(Dm^2n^2)$ [15]. For the application of the OTIFAFr algorithm, we will use an updated version of the C++ utility we previously developed for the IFIMOA and IVIFIMOA algorithms. As we outlined before [2], [16], it is based on a template class which allows us to replace its type with any C++ type or class that implements basic comparison and arithmetic operators. This has allowed us to use the same code for both IFPs (as we will do here) and IVIFPs (as we have done in [16], [21]). The program is command-line based. It expects the following input arguments: a 3-D TIFIM of the experts' evaluations, a matrix of the experts' rating coefficients and a matrix of the weight coefficients of each criterion for each service. The expert evaluations can be given either directly as an index matrix of IFPs, or as a matrix of mark intervals. For the latter case, the first argument of the program must be "interval" followed by the lowest and highest possible mark that an expert can give [21].

B. An Application of OTIFAFr to the Largest Fast Food Restaurant Chain in Bulgaria

In this section, the proposed OTIFAFr model from Sect. III-A is demonstrated with a real case study for choosing a franchisee for the largest fast food restaurant chain in Bulgaria. The optimal problem is defined below: The largest fast food restaurant chain in Bulgaria has given a decision to expand its business through the selection of a franchisee. The franchisor decides to invite a team of 3 experts to evaluate the 3 candidates at a particular moment t_f . The evaluation system consists of 4 groups of criteria: C_1 - owner profitability and business experience level; C_2 - brand marketing and franchise brand development concept; C_3 - opportunities to quickly start a franchise and actively participate in the management of the restaurant and C_4 - restaurant traffic management and parking options, strategic location of the restaurant and successful traffic management around it with provided parking opportunities. Each criteria has priority coefficient as TIFPs pk_{c_j, v_e, t_f} according to their importance from the franchisor's point of view at a current moment t_f . The experts' ratings are defined by TIFP $\{r_1(t_f), r_2(t_f), r_3(t_f)\}$ at a particular moment t_f . In the final ranking we admit only candidates with an overall score higher than $\langle 0.6, 0.01 \rangle$ Now we need to optimally rank the candidates and select the most eligible one.

Solution of the problem:

Step 1. At this step, we create the expert evaluation TIFIM $EV[K, C, E, \{es_{k_i, c_j, d_s}\}]$ with the estimates of the d_s -th expert for the k_i -th candidate by the c_j -th criterion (for $1 \leq i \leq 3, 1 \leq j \leq 4, 1 \leq s \leq 3$) and its form is:

$$\begin{array}{c|cccc} d_1 & c_1 & c_2 & c_3 & c_4 \\ \hline k_1 & \langle 0.4, 0.2 \rangle & \langle 0.3, 0.4 \rangle & \langle 0.7, 0.1 \rangle & \langle 0.3, 0.4 \rangle \\ k_2 & \langle 0.2, 0.7 \rangle & \langle 0.5, 0.3 \rangle & \langle 0.5, 0.4 \rangle & \langle 0.5, 0.3 \rangle \\ k_3 & \langle 0.5, 0.1 \rangle & \langle 0.2, 0.6 \rangle & \langle 0.3, 0.3 \rangle & \langle 0.7, 0.1 \rangle \\ \hline d_2 & c_1 & c_2 & c_3 & c_4 \\ \hline k_1 & \langle 0.5, 0.3 \rangle & \langle 0.2, 0.6 \rangle & \langle 0.8, 0.0 \rangle & \langle 0.4, 0.4 \rangle \\ k_2 & \langle 0.3, 0.7 \rangle & \langle 0.4, 0.4 \rangle & \langle 0.7, 0.1 \rangle & \langle 0.7, 0.0 \rangle \\ k_3 & \langle 0.4, 0.3 \rangle & \langle 0.4, 0.5 \rangle & \langle 0.2, 0.6 \rangle & \langle 0.5, 0.3 \rangle \\ \hline d_3 & c_1 & c_2 & c_3 & c_4 \\ \hline k_1 & \langle 0.2, 0.6 \rangle & \langle 0.3, 0.6 \rangle & \langle 0.5, 0.3 \rangle & \langle 0.5, 0.3 \rangle \\ k_2 & \langle 0.2, 0.7 \rangle & \langle 0.4, 0.5 \rangle & \langle 0.3, 0.5 \rangle & \langle 0.6, 0.1 \rangle \\ k_3 & \langle 0.4, 0.4 \rangle & \langle 0.3, 0.6 \rangle & \langle 0.4, 0.5 \rangle & \langle 0.5, 0.4 \rangle \end{array}$$

Step 2. The rating coefficients of the experts at t_f are: $\{r_1(t_f), r_2(t_f), r_3(t_f)\} = \{(0.7, 0.05), (0.6, 0.05), (0.8, 0.05)\}$.

The TIFIM $V(t_f)[K, C, E, \{v_{k_i, c_j, d_s, t_f}\}]$, which contains the final score of each franchisee at a current moment t_f , is constructed by

$$V(t_f) = r_1 pr_{K,C,d_1,t_f} EV \oplus_{(max,min)} r_2 pr_{K,C,d_2,t_f} EV \oplus_{(max,min)} \oplus_{(max,min)} r_3 pr_{K,C,d_3,t_f} EV(t_f); EV(t_f) := V(t_f) \quad (10)$$

Then we apply the operation $\alpha_{E, \#_q}(EV, h_0) = R[K, C, h_0]$

h_0	c_1	c_2	c_3	c_4
k_1	$\langle 0.3, 0.24 \rangle$	$\langle 0.24, 0.3 \rangle$	$\langle 0.49, 0.05 \rangle$	$\langle 0.4, 0.34 \rangle$
k_2	$\langle 0.18, 0.72 \rangle$	$\langle 0.35, 0.34 \rangle$	$\langle 0.42, 0.15 \rangle$	$\langle 0.48, 0.05 \rangle$
k_3	$\langle 0.35, 0.15 \rangle$	$\langle 0.24, 0.53 \rangle$	$\langle 0.32, 0.34 \rangle$	$\langle 0.49, 0.15 \rangle$

to calculate the aggregated value of the k_i -th applicant about c_j -th criterion at a current moment $h_0 \notin D$, where $\#_q$ is equal to 1, 2 or 3 depending on whether the pessimistic, averaging or optimistic scenarios are chosen. The franchise chain has a requirement for the candidates, so that their total score is not less than a predetermined TIFP $\langle 0.6, 0.01 \rangle$. The level operator (5) is applied to the TIVIFIM B and it is established that all candidates meet this requirement.

Step 3. At this step, a TIFIM $PK(h_0)$ of the weight coefficients of the assessment criterion according to its antecedence is created from the franchisor v_e :

$$PK(h_0)[C, v_e, t_f, \{pk_{c_j, v_e, h_0}\}] = \begin{array}{c|c} h_0 & v_e \\ \hline c_1 & \langle 0.8, 0.1 \rangle \\ c_2 & \langle 0.7, 0.1 \rangle \\ c_3 & \langle 0.5, 0.2 \rangle \\ c_4 & \langle 0.7, 0.1 \rangle \end{array}$$

$$\text{and } B = R \odot_{(\circ,*)} PK = \begin{array}{c|c} h_0 & v_e \\ \hline k_1 & \langle 0.656, 0.0148 \rangle \\ k_2 & \langle 0.661, 0.0136 \rangle \\ k_3 & \langle 0.669, 0.0142 \rangle \end{array}$$

Step 4. The optimistic aggregation operation $\alpha_{K, \#_3}(B, k_0)$ finds that k_3 is the optimal franchisee for the franchise chain of fast food restaurants in Bulgaria v_e with the maximum degree of acceptance (d.a.) 0.669 and the minimum degree of rejection (d.r.) 0.0142 in an optimistic scenario, in a pessimistic scenario – k_1 with the minimum d.a. 0.656 and the maximum d.r. 0.0148. The closest to the average scenario is k_2 with the d.a. 0.661 and the d.r. 0.0136.

Step 5. At this step, we apply the ICRA with $\alpha = 0.8$ and $\beta = 0.10$ over $R(h_0)$. The conclusion is that the evaluation system in the chain is optimized.

The results, obtained from the ICRA application [11], are in the form of IM in $\mu - \nu$ view result matrix:

	c_1	c_2	c_3	c_4
c_1	\perp	$\langle 0.33, 0.48 \rangle$	$\langle 0.62, 0.29 \rangle$	$\langle 0.57, 0.29 \rangle$
c_1	$\langle 0.33, 0.48 \rangle$	\perp	$\langle 0.52, 0.29 \rangle$	$\langle 0.57, 0.19 \rangle$
c_1	$\langle 0.62, 0.29 \rangle$	$\langle 0.52, 0.29 \rangle$	\perp	$\langle 0.67, 0.19 \rangle$
c_1	$\langle 0.57, 0.29 \rangle$	$\langle 0.57, 0.19 \rangle$	$\langle 0.67, 0.19 \rangle$	\perp

Step 6. At last step, the experts' assessments are correct from the point of view of IF logic [6] and their new rating coefficients are equal to $\{(0.82, 0.05), (0.64, 0.05), (0.81, 0.05)\}$.

IV. CONCLUSION

In the study, we have defined the OTIFAFr procedure for selection of the most suitable franchisee over temporal IF evaluations. A software implementation of the proposed algorithm was presented and the decision making procedure

applied for a franchisee selection. In the future, the study will continue with the development of OTIFAFr approach, so that it can be applied over the data, saved in extended TIFIMs [5] and also with software for its implementation.

REFERENCES

- [1] B. Elango, "A bibliometric analysis of franchising research (1988–2017)," *The Journal of Entrepreneurship*, vol. 28 (2), 2019, pp. 223–249.
- [2] D. Mavrov, "An Application for Performing Operations on Two-Dimensional Index Matrices," *Annual of "Informatics" Section, Union of Scientists in Bulgaria*, vol. 10, 2019 / 2020, pp. 66–80.
- [3] D. Mavrov, V. Atanassova, V. Bureva, O. Roeva, P. Vassilev, R. Tsvetkov, D. Zoteva, E. Sotirova, K. Atanassov, A. Alexandrov, H. Tsakov, "Application of Game Method for Modelling and Temporal Intuitionistic Fuzzy Pairs to the Forest Fire Spread in the Presence of Strong Wind," *Mathematics*, vol. 10, 2022, pp. 1280. <https://doi.org/10.3390/mat10081280>
- [4] E. Szmidi, J. Kacprzyk, "Amount of information and its reliability in the ranking of Atanassov's intuitionistic fuzzy alternatives," in: *Rakus-Andersson, E., Yager, R., Ichalkaranje, N., etc. (eds.)*, Recent Advances in Decision Making, SCI, Springer, Heidelberg, vol. 222, 2009, pp. 7–19.
- [5] K. Atanassov, "Index Matrices: Towards an Augmented Matrix Calculus," *Studies in Computational Intelligence*, Springer, vol. 573, 2014.
- [6] K. Atanassov, "On Intuitionistic Fuzzy Sets Theory," *STUDFUZZ*, vol. 283, Springer, Heidelberg, 2012. DOI: 10.1007/978-3-642-29127-2.
- [7] K. Atanassov, D. Mavrov, V. Atanassova, "Intercriteria decision making: a new approach for multicriteria decision making, based on index matrices and intuitionistic fuzzy sets," *Issues in IFSs and Generalized Nets*, vol. 11, 2014, pp. 1–8.
- [8] K. Atanassov, E. Szmidi, J. Kacprzyk, V. Atanassova, "An approach to a constructive simplification of multiagent multicriteria decision making problems via ICRA," *Comptes rendus de l'Academie bulgare des Sciences*, vol. 70 (8), 2017, pp. 1147–1156.
- [9] K. Atanassov, P. Vassilev, O. Roeva, "Level Operators over Intuitionistic Fuzzy Index Matrices," *Mathematics*, vol. 9, 2021, pp. 366.
- [10] L. Zadeh, "Fuzzy Sets," *Information and Control*, vol. 8 (3), 1965, pp. 338–353.
- [11] N. Ikonov, P. Vassilev, O. Roeva, "ICRAData - Software for InterCriteria Analysis," *Int. Journal Bioautomation*, vol. 22, 2018, pp. 1–10.
- [12] P. Hsu, B. Chen, "Developing and Implementing a Selection Model for Bedding Chain Retail Store Franchisee Using Delphi and Fuzzy AHP," *Quality & Quantity*, vol. 41, 2007, pp. 275–290.
- [13] R. J. Kuo, S. C. Chi, S. S. Kao, "A decision support system for selecting convenience store location through integration of FAHP and artificial neural network," *Computers in Industry*, vol. 47 (2), 2002, pp. 199–214.
- [14] S. Fidanova, M. Ganzha, O. Roeva, "InterCriteria analysis of hybrid ant colony optimization algorithm for multiple knapsack problem," in: *Proceedings of the 16th Conference on Computer Science and Information Systems (FedCSIS), Science and intelligence systems*, Sofia, Bulgaria, vol. 25, 2021, pp. 173–180.
- [15] V. Atanassova, O. Roeva, "Computational complexity and influence of numerical precision on the results of ICRA in the decision making process," *Notes on IFSs*, vol. 24 (3), 2018, pp. 53–63.
- [16] V. Traneva, S. Tranev, D. Mavrov, "Interval-Valued Intuitionistic Fuzzy Decision-Making Method using Index Matrices and Application in Outsourcing," in: *Proceedings of the 16th Conference on Computer Science and Information Systems (FedCSIS)*, Sofia, Bulgaria, vol. 25, 2021, pp. 251–255.
- [17] V. Traneva, S. Tranev, M. Stoenchev, K. Atanassov, "Scaled aggregation operations over two- and three-dimensional index matrices," *Soft computing*, vol. 22, 2019, pp. 5115–5120.
- [18] V. Traneva, S. Tranev, *Index Matrices as a Tool for Managerial Decision Making*, Publ. House of the USB; 2017 (in Bulgarian).
- [19] V. Traneva, S. Tranev, "IF Algorithm for Optimal Selection of Franchisees," in: *Kahraman C. (eds) Infus2022*, Lecture Notes in Networks and Systems, vol. 504, Springer, Cham, pp. 632–640, 2022.
- [20] V. Traneva, S. Tranev, "IF Analysis of Variance of Ticket Sales," in: *Kahraman, C. (eds.) INFUS 2020*, Advances in Intelligent Systems and Computing, vol. 1197, Springer, Cham, 2021, pp. 363–340.
- [21] V. Traneva, S. Tranev, D. Mavrov, "Application of an Interval-Valued Intuitionistic Fuzzy Decision-Making Method in Outsourcing Using a Software Program," *Studies in computational intelligence*, Springer, 2023 (in press)

Data Mining Competition

THE topic of this year's data mining competition is the prediction of the costs related to the execution of forwarding contracts. The data sets made available to participants will contain many years of history of orders appearing on the transport exchange, along with details such as the type of order, basic characteristics of the shipped goods (e.g., dimensions, special requirements), as well as the expected route that a driver will have to cover. The task for the competition participants will be to develop a predictive model that will assess the actual cost of individual orders as accurately as possible. Such a model will be used in the future to support Freight Forwarders in selecting profitable contracts.

The sponsor of the competition is Control System Software

– a software company that has been delivering solutions for the Transportation, Spedition, and Logistics industry for 20 years. Attractive prizes are provided for the competition participants:

- 1,000 USD for the winning solution (+ the cost of one FedCSIS 2022 registration)
- 500 USD for the 2nd place solution (+ the cost of one FedCSIS 2022 registration)
- 250 USD for the 3rd place solution (+ the cost of one FedCSIS 2022 registration)

Moreover, Control System Software may award an additional prize (250 USD + the cost of one FedCSIS 2022 registration) to the team that develops the most practical solution to the task.

KnowledgePit Meets BrightBox: A Step Toward Insightful Investigation of the Results of Data Science Competitions

Andrzej Janusz*[†], Dominik Ślęzak*[†]

*Institute of Informatics, University of Warsaw, Warsaw, Poland

[†]QED Software, Warsaw, Poland

Email: {firstname.lastname}@qed.pl

Abstract—We discuss the benefits of integrating the KnowledgePit data science competition platform with the BrightBox technology aimed at diagnostics of machine learning models embedded within complex software systems. We briefly recall the history of international challenges held at KnowledgePit and we also discuss in what sense such technologies as BrightBox can be helpful during the post-challenge analysis. In particular, we show how to combine solutions submitted by the competition participants in order to obtain even more accurate predictions. The discussed functionalities are of significant importance for the sponsors and organizers of data science / machine learning online contests because they support adoption of submissions while designing ultimate solutions of real-world problems.

Index Terms—Data science competitions; machine learning; model stacking; KnowledgePit platform; BrightBox technology

I. INTRODUCTION

KNOWLEDGE Pit¹ is an online platform for organizing data science / machine learning (ML) challenges. Its architecture was first presented in [1] and since then, it has been improving continually. Currently, KnowledgePit puts together the functionalities of a typical competition platform – such as Kaggle² – with additional tools that make it possible for the competition sponsors and organizers to investigate the submitted solutions with respect to their *true* usefulness in the corresponding real-life decision problems. These tools are available thanks to integrating KnowledgePit with BrightBox – the technology developed by QED Software³ for the purpose of assessing the decision models basing on the analysis of mistakes that they are making [2]. In this paper, we discuss one of such functionalities – designed at the border of KnowledgePit and BrightBox – which lets us create better models by mixing solutions acquired from the competition participants.

The paper is organized as follows: In Section II, we recall the main ideas behind KnowledgePit and, as an illustration, we report the history of the KnowledgePit contests held in cooperation with the FedCSIS conference series. Analogously, Section III introduces the main ideas behind BrightBox, with a special emphasis on its contributions into the KnowledgePit’s

functionality. Section IV refers to the data science challenge which was associated with this year’s FedCSIS conference⁴. Besides describing the competition itself, we include here some KnowledgePit-supported visualizations that can be helpful for sponsors and organizers. In Section V, we explain our aforementioned idea of mixing the competition solutions and report the experimental results obtained for the challenge outlined in Section IV. Section VI concludes the paper.

II. THE HISTORY OF KNOWLEDGE PIT

The platforms such as Kaggle attract thousands of data scientists to participate in challenges aiming at solving real-life problems. Such challenges not only address specific problems but often facilitate innovative applications of ML algorithms. On the one hand, they are appealing to those for whom competitive challenges can be a source of new interesting research topics. They can also be an attractive addition to academic courses for students who are interested in practical applications. On the other hand, setting up a public data science competition is a form of outsourcing a given task to the community [3]. It can be beneficial to the sponsors and organizers who set up the contest, as it is an inexpensive approach to solve the problem that they are after [4].

Accordingly, it should not be surprising that the scope of our own platform – KnowledgePit – shifted during the years from organizing smaller, mostly student-focused challenges and projects to international data science competitions. Although KnowledgePit still hosts several student competitions for ML-related university courses every year, the most prestigious events are those prepared for big industry clients and partners, in association with international conferences [5], [6].

One may say that our competitions grew together with recognition of the FedCSIS conferences. Together with the one reported in Section IV [7], there have been already nine challenges held at KnowledgePit in cooperation with FedCSIS. The series started in 2014 with the *AAIA’14 Data Mining Competition: Key Risk Factors for Polish State Fire Service* [8]. Other competition topics included the recognition of firefighters’ activities based on inertial sensor readings [9], predicting seismic activity in coal mines [10], predicting video

This research was co-funded by the Polish National Centre for Research and Development in frame of project MAZOWSZE/0198/19.

¹<https://knowledgepit.ai/>

²<https://www.kaggle.com>

³<https://qed.pl/project/brightbox>

⁴<https://knowledgepit.ai/fedcsis-2022-challenge/>

game winners based on game logs [11], marking hair follicles on microscopic images⁵, predicting win-rates of custom card decks in collectible card games [12], [13], and predicting typical patterns in network device workloads [14]. All of these competitions were highly successful. With more than 1,300 participating teams and several thousands of submitted solutions, they significantly contributed to solving important real-life challenges. They also provided us with a comprehensive survey on the state-of-the-art ML approaches in the related fields, such as time series forecasting [15], feature extraction [16], as well as prediction model ensembling [17].

III. KNOWLEDGEPIT MEETS BRIGHTBOX

During its journey, KnowledgePit had to evolve to fit the needs of our industrial partners. One of the most significant needs has been related to the post-competition analysis of the submitted solutions. With this regard, it was possible to meet the industry expectations thanks to integrating KnowledgePit with the aforementioned BrightBox technology.

As highlighted in Section I, BrightBox is a software technology which assesses decision models (aimed at classification, regression, prediction, etc.) basing on their mistakes (i.e. differences between their outputs and the observed ground truth). Its main application field is in diagnostics of ML models deployed within complex systems [18]. Its methods have deep roots in the theory of rough sets [19]. A diagnosed model is approximated by the rough-set-based surrogate models – the ensembles of so-called approximate decision reducts [20]. Then, particular cases are investigated by looking at their neighborhoods – the groups of other cases that are classified similarly by the surrogate reducts (i.e. objects that fit the same rules induced by reducts [21]). If a mistake that happened for a given case often repeats in its neighborhood, then it is likely that the diagnosed model was not trained correctly for such objects from the beginning. As another example, if the neighborhood is almost empty (i.e.: the diagnosed object seems to be classified by different rules than objects observed before), then BrightBox can conclude that the model seems to find itself in a new situation. Such hints are useful from an operational viewpoint when it comes to rebuilding ML models as parts of the aforementioned complex systems.

Although it was a non-trivial effort to adapt the default settings of BrightBox to the specifics of a data science competition platform (for instance, we had to address different scenarios of the access to the diagnosed models' behavior on the training data), it enabled us to extend basic KnowledgePit's functionalities (such as the analysis of trends in the quality scores or survey-based summaries of commonly applied ML techniques) with in-depth diagnostics of individual submissions. Since BrightBox does not require a direct access to the diagnosed models, it can be applied to construct the above-discussed surrogate models that approximate submitted solutions and allow reasoning about their properties. For example, it allows to approximate feature importance coefficients of

⁵<https://knowledgepit.ai/esensei-challenge/>

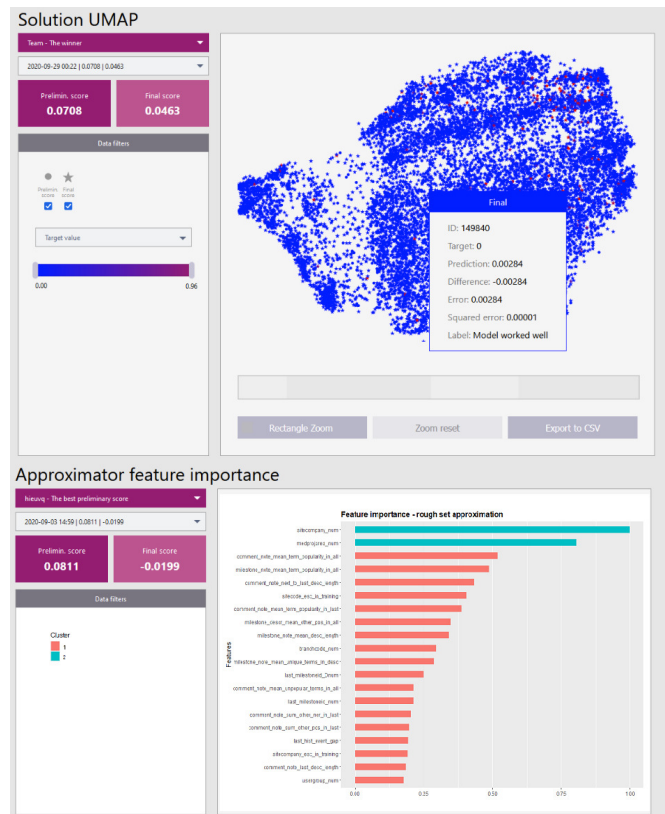


Fig. 1: Exemplary visualizations of one of solutions from challenge [5]. Points on the upper plot correspond to cases from the test data and their color reflects prediction errors. The lower plot shows approximated (using BrightBox) feature importance for the model used to create the analyzed submission. An experimental evaluation showed that the Spearman correlation between the approximated feature importance values and the actual values estimated for the model was ≈ 0.7 .

models used to create the submissions (see Figure 1). It can also provide insightful information on types of errors committed by models, and similarities between solutions submitted by competition participants. Prototype implementations of some selected functionalities provided by the BrightBox technology have been already tested in our previous competitions [5], [22].

IV. FEDCSIS 2022 CHALLENGE

As mentioned in Section II, FedCSIS 2022 hosted the ninth KnowledgePit competition associated with the FedCSIS series. We use this competition as an illustration how KnowledgePit works, as well as a prerequisite for the discussion in Section V. We refer to [7] for more details about this particular contest. The best solutions submitted by its participants are described in [23] (1st place), [24] (2nd), [25] (3rd), and [26] (4th).

The task was to predict the execution costs of so-called forwarding contracts. The data about contracts was provided by the competition sponsor – a company that develops decision

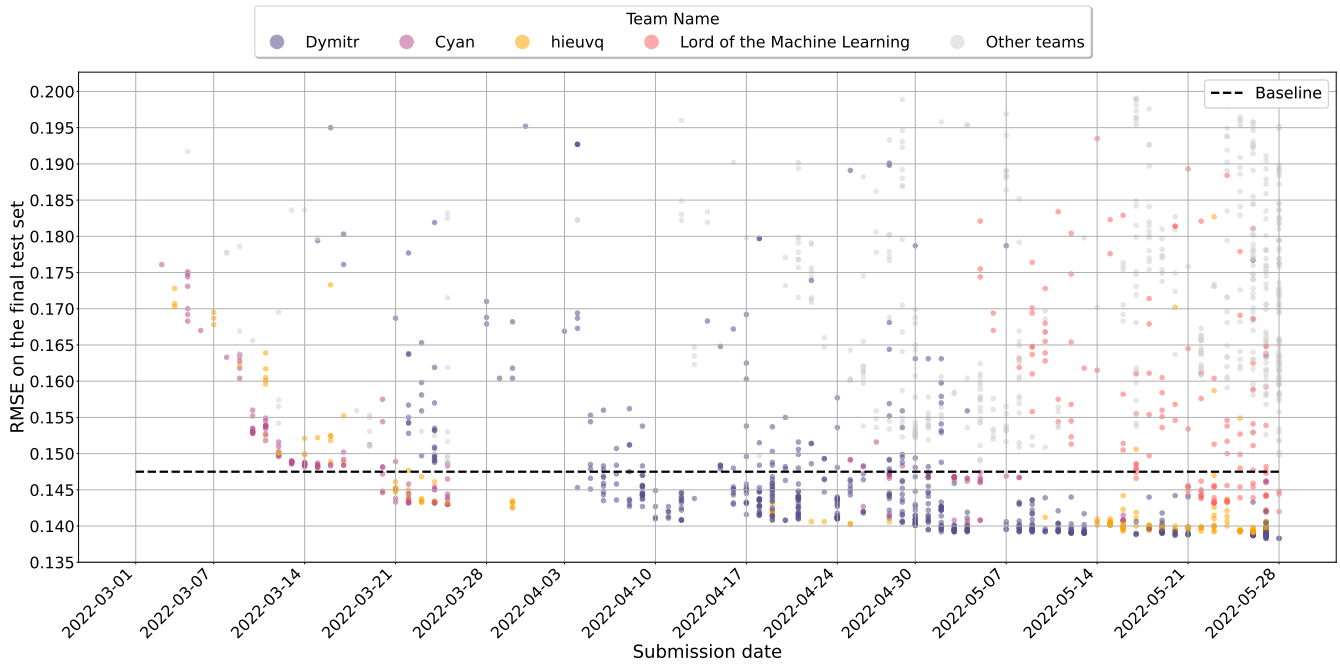


Fig. 2: The final scores of all solutions that reached less than 0.2 RMSE. The most successful teams are specially highlighted.

TABLE I: Top results of the FedCSIS 2022 Challenge [7]. Both preliminary and final scores reflect the RMSE measure.

rank	team name	preliminary	final	#submissions
1	Dymitr [23]	0.1398	0.1383	619
2	Cyan [24]	0.1402	0.1391	181
3	hieuvq [25]	0.1396	0.1407	159
4	Lord of the ML [26]	0.1434	0.1420	147
5	baseline [7]	0.1491	0.1475	-

TABLE II: The RMSE results of the solution stacking methods on the final test set. The simple solution averaging and the weighted solution averaging are included as a reference.

ensemble	final	#solutions
<i>average - best solution of top 5 teams</i>	0.1361	5
<i>weighted average - best solution of top 5 teams</i>	0.1360	5
<i>average - best solution of top 8 teams</i>	0.1366	8
<i>weighted average - best solution of top 8 teams</i>	0.1367	8
<i>LASSO reg. (opt. λ) best solution of each team</i>	0.1344	8
<i>average - top 28 solutions</i>	0.1358	28
<i>weighted average - top 28 solutions</i>	0.1358	28
<i>LASSO reg. (opt. λ) all solutions</i>	0.1339	28

support and optimization systems in the transportation, spedition, and logistics areas. That company was highly interested in a deeper analysis of the submitted solutions with respect to their potential deployment within the designed systems.

For the evaluation purposes, the data was divided into the training set (330,055 historical contracts) and test set (72,452 newer contracts). The data was carefully anonymized, but it

was done in such a way that its analytical value is not lost (see also our other competitions [5]). Given a regression nature of the considered prediction task, we selected one of typical evaluation measures – the root mean square error (RMSE). However, let us note that sometimes specifying a measure that truly reflects a real-life decision problem is not easy [27].

The competition was conducted in a standard way including: (i) the online preliminary evaluation on an unknown subset of the test data (the participants submit solutions for the full test set but preliminary evaluation is done on a subset which is unknown to them), (ii) the associated public leaderboard (only preliminary results are shown during the competition), and (iii) the final evaluation on the full test set (which is calculated after the competition is closed, after the participants select their final solutions to be evaluated, and only if they submit to KnowledgePit the reports that describe their solutions).

Table I displays the top results. Let us note that the preliminary and final rankings are different. Actually, the solution at the 3rd final place would be the best one if only preliminary scores were taken into account. Such cases are specially worth investigating with the usage of the BrightBox technology.

Table I shows also the baseline solution – the model that we prepared by ourselves prior to the contest’s beginning [7]. In this challenge, only four teams exceeded the baseline score (though sometimes it may be even less, see e.g. [14]). Figure 2 displays the history of all submissions of those teams.

V. MIXING THE SOLUTIONS

Due to a large number of submitted solutions and the fact that most of the best-performing teams used some advanced ensembling methods to compute their final predictions [23],

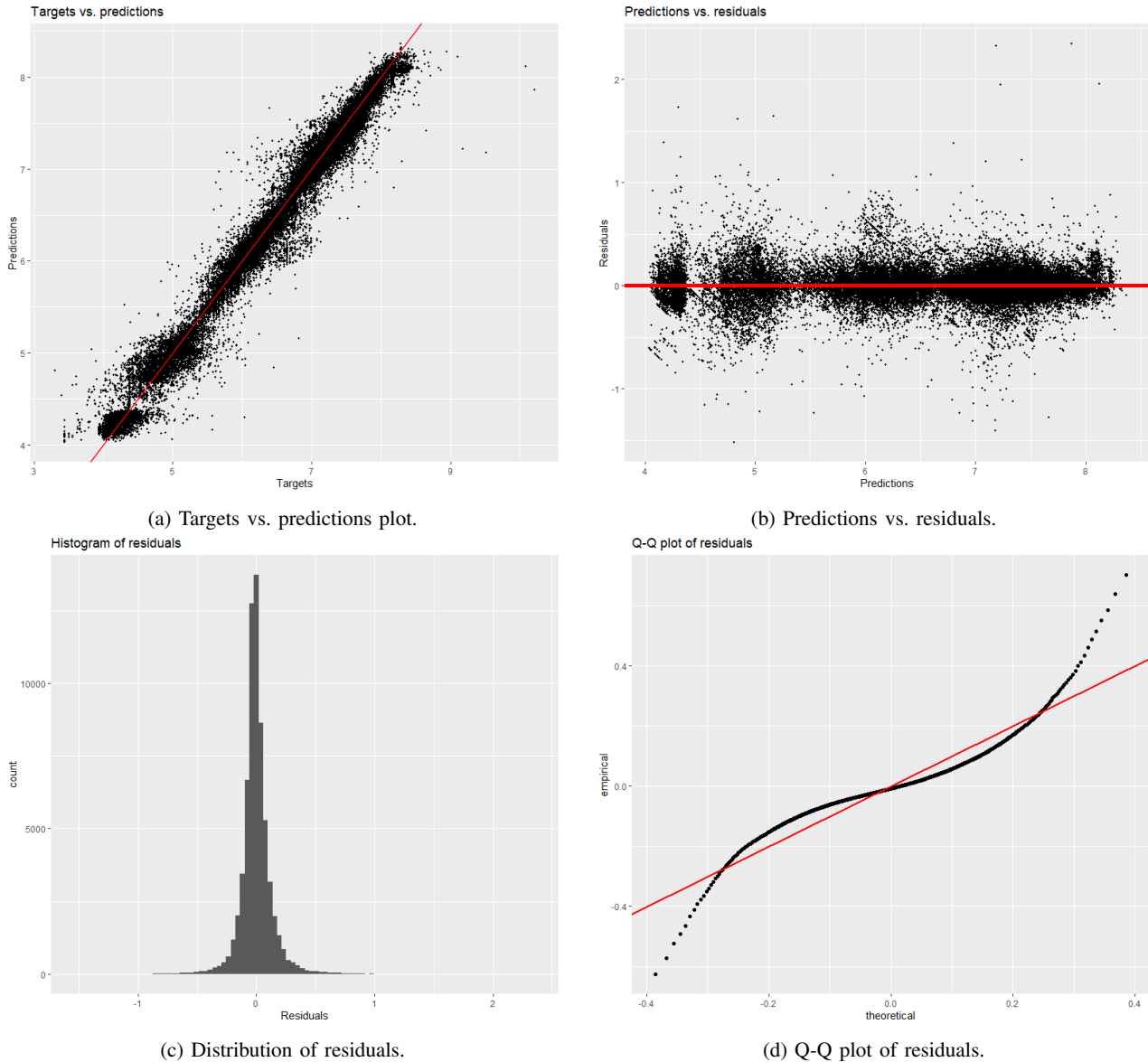


Fig. 3: Visual quality analysis of the ensemble model obtained using the considered solution mixing methodology. The top-left plot shows the relation between ground truths and predictions for test cases. The top-right scatter plot visualizes residuals with regard to predictions. The bottom-left plot shows the distribution of residuals on test cases, and the bottom-right plot shows the relation between theoretical quantiles of the Gaussian distribution and the empirical quantiles of the residuals.

[24], [25], [26], we decided to check how accurate the predictions could be if we mix submissions from different teams. In this way, not only could we verify which solutions are the most influential and potentially worth deploying, but also we could get a better insight into where the limit to prediction quality is when working with this type of the data.

To create the ensemble of submissions, we used the post-competition data analytics tools provided by KnowledgePit thanks to its integration with BrightBox (see also [22]). We compared the results of the LASSO regression-based model stacking [28] to simple model averaging methods. We also

investigated two ensembling scenarios. In the first one, we used all correctly formatted submissions. In the second one, we restrained the submissions to those with the best score in the preliminary evaluation from each team. In both cases, we trained the logistic regression model with LASSO regularization on predictions for the competition's preliminary evaluation set. Then, we verified the performance of each ensemble on the same final test set as all individual submissions.

We tuned the regularization parameter λ of the LASSO regression using the cross-validation technique, following the

guidelines taken from the `glmnet` package⁶. We chose λ for which the validation loss was lowest. Table II shows the results for the obtained ensembles. As a reference, we include the results obtained using simple averaging of the best solutions from top 5 teams (solutions [23], [24], [25], [26] and the baseline model), and the results of weighted averaging in which the weights correspond to the preliminary scores. We also include results obtained for the simple averaging, and weighted averaging when the number of used solutions corresponds to the optimal selection of the λ value.

The ensemble that achieved the best RMSE was trained on all submissions. It had non-zero coefficients for 28 solutions submitted by nine different teams. All of those coefficients were positive. The highest total impact on predictions had the submissions of *hieuwq* (0.321), followed by *Lord of the ML* (0.189), *Cyan* (0.156), and *Dymitr* (0.151). The combined impact of the remaining teams was less than 0.187.

We analyzed residuals of the resulting predictions. Figure 3 depicts their distribution across the test data. The top-left plot 3a shows the relation between ground-truth targets and predictions for test cases from the FedCSIS 2022 Challenge. It can be seen that they are aligned along the diagonal (the red line) with relatively few outlying cases. Similarly, in the top-right part of Figure 3 there is a scatter plot of residuals with regard to predictions of the ensemble. It shows that the residuals are not evenly distributed and there are a few prediction ranges with slightly larger magnitude (variance) of prediction errors. Both of those plots also show that the ensemble is nearly unbiased. This observation was confirmed – the mean difference between the ground truths and the predictions is nearly zero, i.e., -0.00027 . However, the distribution of residuals is not Gaussian. Such a hypothesis was rejected with high confidence using the Shapiro-Wilk test⁷. This can also be seen in Figures 3c and 3d. The distribution of residuals has long tails and the frequency of high error values is far from the theoretical quantile values of the Gaussian distribution. This observation suggests that the considered ensemble model could be further improved. Herein, a thorough BrightBox-based investigation of the morphology of mistakes made by each of the aforementioned 28 solutions can be helpful.

VI. CONCLUSIONS

We discussed the idea of organizing online data science competitions, with some examples taken from our own experience. We referred to KnowledgePit – our competition platform which, besides standard functionalities, includes some advanced analytical and visualization tools. We paid a special attention to the BrightBox technology which is used by KnowledgePit to approximate and diagnose solutions submitted by the competition participants. One of the methods which can be used within the resulting KnowledgePit-BrightBox environment, refers to mixing different solutions together, which leads toward more efficient ML models, as well as additional insights with regard to particular submissions.

⁶<https://cran.r-project.org/web/packages/glmnet/glmnet.pdf>

⁷https://en.wikipedia.org/wiki/Shapiro-Wilk_test

Consequently, in order to provide even more value to the future competition sponsors, we are extending KnowledgePit’s functionalities to utilize a broader range of XAI tools for the purpose of advanced post-challenge research. We also believe that such functionalities can be useful from an education perspective, whereby KnowledgePit may be considered as a platform for assessing and improving the data science competencies at universities, as well as in companies.

We also plan to continue integration of KnowledgePit with BrightBox. In particular, we are going to develop better tools for inter-competition analysis of individual platform users. Such functionality would help us to monitor the progress and skill development of participants. It could also provide value to our industrial partners who often look for potential skilled employees. In this context, KnowledgePit could be used by our partners as a tool that facilitates the recruitment of researchers for projects related to the competition topics, and for the evaluation of job candidates. Actually, QED Software is already using KnowledgePit for such evaluations.

REFERENCES

- [1] A. Janusz, D. Ślęzak, S. Stawicki, and M. Rosiak, “Knowledge Pit – A Data Challenge Platform,” in *Proceedings of the 24th International Workshop on Concurrency, Specification and Programming, Rzeszów, Poland, September 28-30, 2015*, ser. CEUR Workshop Proceedings, Z. Suraj and L. Czaja, Eds., vol. 1492. CEUR-WS.org, 2015, pp. 191–195. [Online]. Available: http://ceur-ws.org/Vol-1492/Paper_18.pdf
- [2] A. Janusz, A. Zalewska, and D. Ślęzak, “Introducing Approximation-based Model Diagnostics into KnowledgePit – A Platform for Organizing Data Mining Challenges,” in *Book of Abstracts, the 19th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 2022), Milan, Italy, July 11-15, 2022*, S. Boffa, D. Ciucci, I. Couso, J. Medina, and D. Ślęzak, Eds., 2022, pp. 19–20. [Online]. Available: <https://ipmu2022.disco.unimib.it/wp-content/uploads/sites/86/2022/07/IPMU22-book-of-abstract.pdf>
- [3] J. L. Zimmermann, “Data Competitions: Crowdsourcing with Data Science Platforms,” in *The Machine Age of Customer Insight*. Emerald Publishing Limited, 2021, pp. 183–197. [Online]. Available: <https://doi.org/10.1108/978-1-83909-694-520211017>
- [4] C. Tauchert, P. Buxmann, and J. Lambinus, “Crowdsourcing Data Science: A Qualitative Analysis of Organizations’ Usage of Kaggle Competitions,” in *Proceedings of the 53rd Hawaii International Conference on System Sciences, HICSS 2020, Maui, Hawaii, USA, January 7-10, 2020*, 2020, pp. 1–10. [Online]. Available: <https://doi.org/10.24251/HICSS.2020.029>
- [5] A. Janusz, G. Hao, D. Kałuża, T. Li, R. Wojciechowski, and D. Ślęzak, “Predicting Escalations in Customer Support: Analysis of Data Mining Challenge Results,” in *2020 IEEE International Conference on Big Data (IEEE BigData 2020), Atlanta, GA, USA, December 10-13, 2020*, X. Wu, C. Jermaine, L. Xiong, X. Hu, O. Kotevska, S. Lu, W. Xu, S. Aluru, C. Zhai, E. Al-Masri, Z. Chen, and J. Saltz, Eds., 2020, pp. 5519–5526. [Online]. Available: <https://doi.org/10.1109/BigData50022.2020.9378024>
- [6] A. Janusz, D. Kałuża, A. Chadzyńska-Krasowska, B. Konarski, J. Holland, and D. Ślęzak, “IEEE BigData 2019 Cup: Suspicious Network Event Recognition,” in *2019 IEEE International Conference on Big Data (IEEE BigData), Los Angeles, CA, USA, December 9-12, 2019*, C. K. Baru, J. Huan, L. Khan, X. Hu, R. Ak, Y. Tian, R. S. Barga, C. Zaniolo, K. Lee, and Y. F. Ye, Eds., 2019, pp. 5881–5887. [Online]. Available: <https://doi.org/10.1109/BigData47090.2019.9005668>
- [7] A. Janusz, A. Jamiołkowski, and M. Okulewicz, “Predicting the Costs of Forwarding Contracts: Analysis of Data Mining Competition Results,” in *Proceedings of the 2022 Federated Conference on Computer Science and Intelligence Systems, Sofia, Bulgaria, September 4-7, 2022*, ser. Annals of Computer Science and Information Systems, M. Ganzha, M. Paprzycki, and D. Ślęzak, Eds., vol. 30, 2022.

- [8] A. Janusz, A. Krasuski, S. Stawicki, M. Rosiak, D. Ślęzak, and H. S. Nguyen, "Key Risk Factors for Polish State Fire Service: A Data Mining Competition at Knowledge Pit," in *Proceedings of the 2014 Federated Conference on Computer Science and Information Systems, Warsaw, Poland, September 7-10, 2014*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 2, 2014, pp. 345–354. [Online]. Available: <https://doi.org/10.15439/2014F507>
- [9] M. Meina, A. Janusz, K. Rykaczewski, D. Ślęzak, B. Celmer, and A. Krasuski, "Tagging Firefighter Activities at the Emergency Scene: Summary of AAI'A'15 Data Mining Competition at Knowledge Pit," in *2015 Federated Conference on Computer Science and Information Systems, FedCSIS 2015, Łódź, Poland, September 13-16, 2015*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 5, 2015, pp. 367–373. [Online]. Available: <https://doi.org/10.15439/2015F426>
- [10] A. Janusz, D. Ślęzak, M. Sikora, and Ł. Wróbel, "Predicting Dangerous Seismic Events: AAI'A'16 Data Mining Challenge," in *Proceedings of the 2016 Federated Conference on Computer Science and Information Systems, FedCSIS 2016, Gdańsk, Poland, September 11-14, 2016*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 8, 2016, pp. 205–211. [Online]. Available: <https://doi.org/10.15439/2016F560>
- [11] A. Janusz, T. Tajmajer, and M. Świechowski, "Helping AI to Play Hearthstone: AAI'A'17 Data Mining Challenge," in *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems, FedCSIS 2017, Prague, Czech Republic, September 3-6, 2017*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 11, 2017, pp. 121–125. [Online]. Available: <https://doi.org/10.15439/2017F573>
- [12] A. Janusz, Ł. Grad, and M. Grzegorowski, "Clash Royale Challenge: How to Select Training Decks for Win-rate Prediction," in *Proceedings of the 2019 Federated Conference on Computer Science and Information Systems, FedCSIS 2019, Leipzig, Germany, September 1-4, 2019*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 18, 2019, pp. 3–6. [Online]. Available: <https://doi.org/10.15439/2019F365>
- [13] A. Janusz, T. Tajmajer, M. Świechowski, Ł. Grad, J. Puczniewski, and D. Ślęzak, "Toward an Intelligent HS Deck Advisor: Lessons Learned from AAI'A'18 Data Mining Competition," in *Proceedings of the 2018 Federated Conference on Computer Science and Information Systems, FedCSIS 2018, Poznań, Poland, September 9-12, 2018*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 15, 2018, pp. 189–192. [Online]. Available: <https://doi.org/10.15439/2018F386>
- [14] A. Janusz, M. Przyborowski, P. Biczuk, and D. Ślęzak, "Network Device Workload Prediction: A Data Mining Challenge at Knowledge Pit," in *Proceedings of the 2020 Federated Conference on Computer Science and Information Systems, FedCSIS 2020, Sofia, Bulgaria, September 6-9, 2020*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 21, 2020, pp. 77–80. [Online]. Available: <https://doi.org/10.15439/2020F159>
- [15] M. Züfle and S. Kounev, "A Framework for Time Series Preprocessing and History-based Forecasting Method Recommendation," in *Proceedings of the 2020 Federated Conference on Computer Science and Information Systems, FedCSIS 2020, Sofia, Bulgaria, September 6-9, 2020*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 21, 2020, pp. 141–144. [Online]. Available: <https://doi.org/10.15439/2020F101>
- [16] P. Przybyszewski, S. Dziwiątkowski, S. Jaszczur, M. Śmiech, and M. S. Szczuka, "Use of Domain Knowledge and Feature Engineering in Helping AI to Play Hearthstone," in *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems, FedCSIS 2017, Prague, Czech Republic, September 3-6, 2017*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 11, 2017, pp. 143–148. [Online]. Available: <https://doi.org/10.15439/2017F567>
- [17] M. Grzegorowski, "Massively Parallel Feature Extraction Framework Application in Predicting Dangerous Seismic Events," in *Proceedings of the 2016 Federated Conference on Computer Science and Information Systems, FedCSIS 2016, Gdańsk, Poland, September 11-14, 2016*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 8, 2016, pp. 225–229. [Online]. Available: <https://doi.org/10.15439/2016F90>
- [18] A. Gosiewska and P. Biecek, "Auditor: An R Package for Model-Agnostic Visual Validation and Diagnostics," *The R Journal*, vol. 11, no. 2, p. 85, 2019. [Online]. Available: <https://doi.org/10.32614/rj-2019-036>
- [19] A. Skowron and D. Ślęzak, "Rough Sets Turn 40: From Information Systems to Intelligent Systems," in *Proceedings of the 2022 Federated Conference on Computer Science and Intelligence Systems, Sofia, Bulgaria, September 4-7, 2022*, ser. Annals of Computer Science and Information Systems, M. Ganzha, M. Paprzycki, and D. Ślęzak, Eds., vol. 30, 2022.
- [20] S. Stawicki, D. Ślęzak, A. Janusz, and S. Widz, "Decision Bireducts and Decision Reducts – A Comparison," *International Journal of Approximate Reasoning*, vol. 84, pp. 75–109, 2017. [Online]. Available: <https://doi.org/10.1016/j.ijar.2017.02.007>
- [21] J. W. Grzymała-Busse, "Rule Induction," in *Data Mining and Knowledge Discovery Handbook, 2nd ed*, O. Maimon and L. Rokach, Eds. Springer, 2010, pp. 249–265. [Online]. Available: https://doi.org/10.1007/978-0-387-09823-4_13
- [22] M. Matraszek, A. Janusz, M. Świechowski, and D. Ślęzak, "Predicting Victories in Video Games – IEEE BigData 2021 Cup Report," in *2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, December 15-18, 2021*, Y. Chen, H. Ludwig, Y. Tu, U. M. Fayyad, X. Zhu, X. Hu, S. Byna, X. Liu, J. Zhang, S. Pan, V. Papalexakis, J. Wang, A. Cuzzocrea, and C. Ordonez, Eds., 2021, pp. 5664–5671. [Online]. Available: <https://doi.org/10.1109/BigData52589.2021.9671650>
- [23] D. Ruta, M. Liu, L. Cen, and Q. H. Vu, "Diversified Gradient Boosting Ensembles for Prediction of the Cost of Forwarding Contracts," in *Proceedings of the 2022 Federated Conference on Computer Science and Intelligence Systems, Sofia, Bulgaria, September 4-7, 2022*, ser. Annals of Computer Science and Information Systems, M. Ganzha, M. Paprzycki, and D. Ślęzak, Eds., vol. 30, 2022.
- [24] H. Xiao, Y. Liu, D. Du, and Z. Lu, "An Approach for Predicting the Costs of Forwarding Contracts using Gradient Boosting," in *Proceedings of the 2022 Federated Conference on Computer Science and Intelligence Systems, Sofia, Bulgaria, September 4-7, 2022*, ser. Annals of Computer Science and Information Systems, M. Ganzha, M. Paprzycki, and D. Ślęzak, Eds., vol. 30, 2022.
- [25] Q. H. Vu, L. Cen, D. Ruta, and M. Liu, "Key Factors to Consider when Predicting the Costs of Forwarding Contracts," in *Proceedings of the 2022 Federated Conference on Computer Science and Intelligence Systems, Sofia, Bulgaria, September 4-7, 2022*, ser. Annals of Computer Science and Information Systems, M. Ganzha, M. Paprzycki, and D. Ślęzak, Eds., vol. 30, 2022.
- [26] S. Pioroński and T. Górecki, "Using Gradient Boosting Trees to Predict the Costs of Forwarding Contracts," in *Proceedings of the 2022 Federated Conference on Computer Science and Intelligence Systems, Sofia, Bulgaria, September 4-7, 2022*, ser. Annals of Computer Science and Information Systems, M. Ganzha, M. Paprzycki, and D. Ślęzak, Eds., vol. 30, 2022.
- [27] M. Aché, A. Janusz, K. Żbikowski, D. Ślęzak, M. Kryszkiewicz, H. Rybiński, and P. Gawrysiak, "ISMIS 2017 Data Mining Competition: Trading Based on Recommendations," in *Foundations of Intelligent Systems - 23rd International Symposium, ISMIS 2017, Warsaw, Poland, June 26-29, 2017, Proceedings*, ser. Lecture Notes in Computer Science, M. Kryszkiewicz, A. Appice, D. Ślęzak, H. Rybiński, A. Skowron, and Z. W. Raś, Eds., vol. 10352. Springer, 2017, pp. 697–707. [Online]. Available: https://doi.org/10.1007/978-3-319-60438-1_68
- [28] T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction (2nd Edition)*. Springer, 2009.

Predicting the Costs of Forwarding Contracts: Analysis of Data Mining Competition Results

Andrzej Janusz^{*†}, Antoni Jamiołkowski[†], Michał Okulewicz^{‡§}

^{*}Institute of Informatics, University of Warsaw, Warsaw, Poland

[†]QED Software, Warsaw, Poland

[‡]Control System Software, Sopot, Poland

[§]Faculty of Mathematics and Information Science, Warsaw University of Technology, Warsaw, Poland

Abstract—We discuss the international competition *FedCSIS 2022 Challenge: Predicting the Costs of Forwarding Contracts* that was organized in association with the FedCSIS conference series at the KnowledgePit platform. We explain the scope and outline the results obtained by the most successful teams.

Index Terms—Data mining competitions; costs of forwarding contracts; KnowledgePit platform

I. INTRODUCTION

TRANSPORTATION and logistics are among the most influential sectors in the global economy. It is their capacity that determines whether the commodities will reach the end customers. Moreover, the cost of transportation has a direct impact on the prices of essential goods. According to the European Union’s Science Hub¹, production expenses among European Union companies consist of up to 15% of transportation and warehousing fees. Hence, appropriate decisions of freight forwarders – individuals involved in the overall arrangement of transportation services – remain vital. The *FedCSIS 2022 Challenge* is an attempt to address this issue.

Freight forwarders commonly use their expert intuition to decide whether to accept or reject a contract. It allows them to arrive at accurate choices, though underneath it is a complex process. The contract cost is affected by, but not limited to, the size of transported goods, their weight, fuel prices and, most importantly, the contract route. Various countries are likely to have significantly different transit-related costs. Furthermore, outlier contracts such as hazardous freight or those requiring additional safeguarding are particularly hard to handle. The motivation for the competition was the presumption that machine learning (ML) can make efficient use of this data.

The paper is organized as follows: Section II reviews the literature on ML in freight forwarding tasks. Section III outlines the objective of the competition and gives some details about the data. Section IV describes the baseline solution that we prepared for the competition purposes. Section V reports the winning solutions. Section VI concludes the paper.

¹This research was partially supported by the Polish National Centre for Research and Development, grant no. POIR.01.01.01-00-2050/20, application track 6/1.1.1/2020 – 2nd round.

²https://joint-research-centre.ec.europa.eu/scientific-activities-z/transport-sector-economic-analysis_en

II. ANALYSIS OF RELATED LITERATURE

The income of the freight forwarder is largely based on a commission from the profit of successfully finding transportation services for the cargo that needed moving. Freight forwards operate on online freight exchanges, such as Timocom², Trans.eu³ or Teleroute⁴, seeking the most profitable contracts for which they can find a transportation service. The accurate transportation cost prediction is one of the key problems that need to be solved by freight forwarders to be successful. The importance of managing forwarders’ information is discussed in [1], while [2] analyzes other factors impacting the financial effectiveness of managing transportation logistics.

An interesting topic is the prediction of the future freight demand [3], which would enable freight forwarders to balance risk and expected income. Other ML-based approaches focus on finding estimated time of arrival (ETA) or predict fuel consumption. In [4] a random forest is used to predict ETA for intermodal transportation (i.e. including sea and/or railway transport). In [5], [6] random forests and support vector machines are also used to predict fuel consumption in order to monitor and prevent fuel fraud. A recent review [7] summarizes the aspects of freight forwarding and transportation, whereby ML approaches have been utilized up to now.

Meanwhile, factors other than time and fuel consumption influencing transportation costs, especially in data-driven ML approaches, have not been thoroughly studied. However, [8], [9] propose expert models to calculate such transportation costs. Moreover, [10] proposes a statistical model and analyzes the impact of various factors on the estimated cost. While [11] also takes into account risk factors for the ocean transportation costs. Finally, [12] tried to solve the problem of cost prediction using artificial neural networks. However, as in most of the mentioned studies, these results were based only on small data sets or expert surveys. We believe that providing research community with a more comprehensive data will prove crucial for finding new factors that impact the accuracy of predicting transportation cost for forwarding contracts.

III. FEDCSIS 2022 CHALLENGE OUTLINE

The challenge was launched at the KnowledgePit platform⁵ on March 1, 2022, and the submission system was opened until

²<https://www.timocom.co.uk/smart-logistics-system/freight-exchange>

³<https://www.trans.eu/en/carriers/>

⁴<https://teleroute.com/en-en/>

⁵<https://knowledgepit.ai/fedcsis-2022-challenge/>

May 27, 2022. We refer to [13] for more information about KnowledgePit, as well as about the previous KnowledgePit competitions that have quite a long tradition at FedCSIS.

The data used this year was provided by the competition sponsor, i.e. Control System Software – a Polish software company that is specializing in solutions for the Transportation, Spedition, and Logistics industry. The task for participants was to predict execution costs of forwarding contracts described in the available test data. An accurate prediction model for this task could be used in future to support freight forwarders.

A. Data preparation

The data sets that were made available in our competition describe an over six-year history of contracts accepted by a large Polish transportation company. The main data was composed of two separate tables. The first one contained basic information about the contracts, and the second one described the main sections of the planned routes associated with each contract. The first column in both tables, i.e. *id_contract*, stored identifiers that allow matching records between them. Additionally, the second column in the first table (the main data file), i.e. *expenses*, contained information about the actual prediction target values. A short description of the remaining data attributes from both tables was also made available in separate files. Finally, an additional data table containing historical wholesale prices of fuel was provided.

Since the data came from a real transportation company, all sensitive information had to be scrambled prior to publishing. All identifiers were removed or encoded by random strings. Geo-location data related to key points on the routes was modified. Instead of original values, we used the Nominatim service⁶ to generate coordinates of the central points in the corresponding post code areas. In the published data, the original geographical coordinates are changed into the generated ones. Some of the characteristics of trucks and trailers were transformed into indicators. Finally, the fuel prices and the target values (the contract execution costs) were rescaled.

For the purpose of the evaluation, the data was divided into separate training and test data tables. The training data contained approximately five-year history of the accepted contracts, and the test set was composed of the data collected in the last year (between Nov. 1, 2020 and Nov. 23, 2021). In total, training data stored information about 330,055 contracts described by 36 attributes, and the total number of route parts was 1,189,654 (the route data table had 60 attributes). The empirical distribution of the target expenses looked like a mixture of a few Gaussians, with the mean value 6.3735 and standard deviation 1.059. The histogram of target values is presented on Figure 1. The test data contained 72,452 contracts, and the corresponding route data consisted of 325,222 entries.

B. Evaluation procedure

The evaluation procedure for our competition was typical to challenges held at KnowledgePit [14], [15]. Competitors submitted solutions as text files with each line containing a single prediction for the corresponding test instance. The quality of submissions was evaluated online. We used RMSE as the

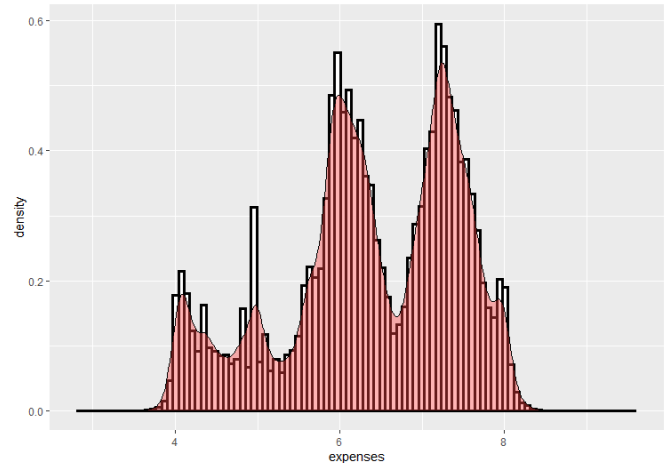


Fig. 1: The target expenses values in the training data set.

error measure. The preliminary score of each submission was computed on a small subset of the test records. Approximately 10% of test data was used for the preliminary evaluation. The best preliminary result for each team was published on the public leaderboard. The final evaluation was performed after the competition's completion using the remaining part of the test instances. Those results were then published online too. Only the teams which submitted a short report describing their approach were qualified for the final evaluation.

IV. COMPETITION BASELINE

To give a reference to competitors, we prepared and submitted at the very beginning predictions of our own baseline model. To construct it, we first analyzed the available data and identified features that could be useful. We divided the data preprocessing task into stages. At each stage, we extracted different types of features describing the contracts from the available data tables. This part of the model preparation process (i.e. feature extraction [16], [17]) proved to be crucial to the performance of the resulting prediction model.

Firstly, we processed the main data table. After consulting with domain experts, we identified categorical features that could have predictive value. For each of such features, we narrowed its set of possible values to those which appeared in at least 1% of training data. All other non-missing values were changed to *other*. After this transformation, we used two types of encoding. The selected features were one-hot encoded. Additionally, we created a numeric version of the categorical features by transforming each value into the mean *expenses* of contracts from the training data with that value. Overall, we applied this transformation to features *id_payer*, *id_currency*, *direction*, *load_size_type*, *contract_type*, *id_service_type*, *first_load_country*, *last_unload_country*, *route_start_country*, *route_end_country*, *prim_train_line*, *prim_ferry_line*, *route_start_month*, and *route_end_month*.

In the second stage, data from the route tables was processed. Again, the filtration of rare categorical values was performed. After that, for each contract we performed projections of aggregated *km*, *km_haversine*, and *kg_current* values on the values of selected categorical features. We added those

⁶<https://nominatim.org/>

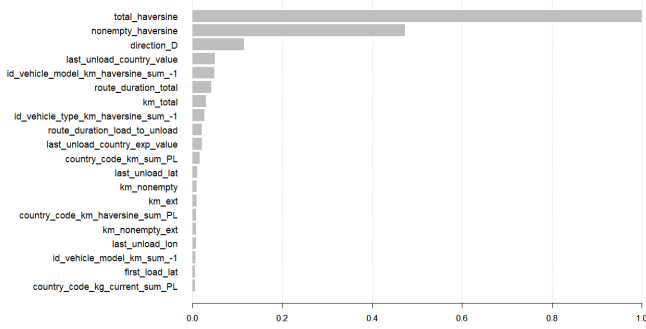


Fig. 2: Estimated feature importance for the baseline model.

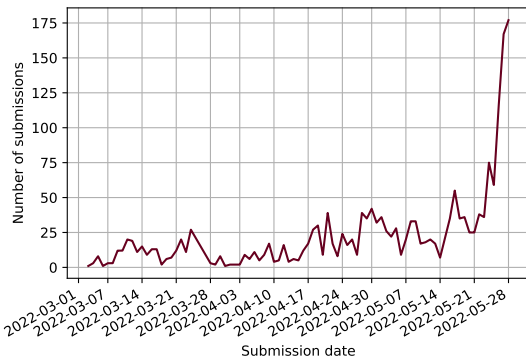


Fig. 3: Activity of participants by means of daily submissions.

projections as new features describing individual contracts. We also constructed a few dozen of other auxiliary features, such as the total number of steps without a load, the number of steps performed by external contractors, the average ratio between the current weight and the length of the route part, etc.

Finally, we added time features (e.g. month number, quarter, whether the contract will be executed through the week-end), fuel prices at the route beginning, and truck features (e.g. average size of the required trucks during the route). In total, we defined 585 numeric features describing the contracts.

We constructed the prediction model using the XGBoost library [18]. We did not focus on the hyperparameter tuning. We used a small portion of the training data as a validation set and experimentally checked several settings. The final model was heavily regularized, i.e. the learning rate was set to 0.01, $\alpha = \lambda = 1$, and subsampling was used on both instances and columns. The total number of used trees was 2,500, and the maximum depth of trees was set to 8. Figure 2 shows the estimation of feature importance in the resulting model. In the competition, our model had the fifth score with the preliminary RMSE value 0.1491 and the final result 0.1475.

V. COMPETITION RESULTS

The challenge was taken up by 130 teams from 24 countries. The teams came e.g. from Poland (76), India (14), and the USA (4). There were 1,927 solutions submitted. Figure 3 shows activity of competitors expressed in terms of the number of daily submissions. It shows that the number of daily

TABLE I: Top 10 final results of the FedCSIS 2022 Challenge.

Rank	Team name	Preliminary	Final score	#subs
1	Dymitr	0.1398	0.1383	619
2	Cyan	0.1402	0.1391	181
3	hieuvq	0.1396	0.1407	159
4	Lord of the ML	0.1434	0.1420	147
5	baseline	0.1491	0.1475	-
6	kubapok	0.1502	0.1494	32
7	DeepIf	0.1500	0.1498	28
8	Stan	0.1529	0.1519	131
9	Artur Budzyński	0.1549	0.1520	45
10	Nindza Zhelki	0.1567	0.1573	36
...

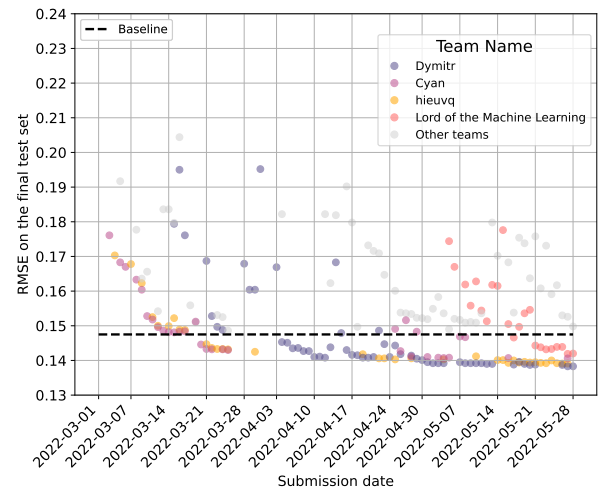


Fig. 4: RMSE (final test set) of solutions of the best teams.

submissions remained stable in the first half of the competition. Teams increased their activity in the second half, reaching the peak in the last week. Solutions submitted during the last 3 days of the competition account for nearly 24% of all solutions. This shows that the competition between participants continued until the last moment. In Table I, we present the final ranks, scores, and the number of submissions of the best performing teams. Figure 4 follows with more details about the teams that eventually managed to exceed our baseline solution.

As in previous KnowledgePit competitions, feature extraction was a crucial step. The solutions of the best-performing teams were preceded by in-depth data analysis and processing. The well-established approach of a feature selection preceded by a feature generation was the most common. Feature generation methods ranged from simple statistics to a manual selection of feature combinations and regex-based information extraction from text describing temperature requirements.

Every team that exceeded the baseline used gradient boosting methods (XGBoost [18], LightGBM [19] or CatBoost [20]) and model ensemble techniques. The winners’ solution puts emphasis on choosing suitable ensemble methods and model diversification, whereby their two proposed approaches focus on the diversity of the models’ hyperparameters and the level of disagreement of the models’ output. Both approaches were used in the final solution. The decision of their final model ensemble was the average of the models’ predictions on two

subsets with different features. The runner-up team used the model stacking approach (i.e. ridge regression trained on top of gradient boosted trees' outputs). On the other hand, the team that finished third conducted a forecast post-processing that aimed to predict a trend in the contract costs. That forecast was used to adjust the predictions of the final model.

VI. CONCLUSIONS

We presented an international data mining competition related to a vital problem in the transportation and logistic industry, i.e. predicting the execution costs of forwarding contracts accepted by a freight company. We described the competition scope and available data sets, and we proposed a baseline model for the task. We also discussed the most successful solutions proposed by the participants.

The competition was a successful event, with 130 registered teams from 24 countries. The most accurate solutions were largely dominated by gradient boosting models implemented in popular libraries, such as XGBoost, and LightGBM. They were typically combined with feature extraction techniques in the data preprocessing phase. Moreover, a few teams decided to mix several models trained on different parts of data, and their final solutions were generated using an ensemble.

Reducing transportation expenses by selecting optimal contracts, or by identifying the most costly factors can decrease the price of production of many goods including those purchased on a daily basis. We believe that our competition contributed to the discussion on the estimation of forwarding contract costs. The solutions developed by participants, and outcomes of future research may pronouncedly influence the decision-making process of transportation and logistics companies. By providing the research community with a large-scale data set, we hope to accelerate the advances in this area.

REFERENCES

- [1] E.-S. Lee and D.-W. Song, "Knowledge Management in Freight Forwarding as a Logistics Intermediator: Model and Effectiveness," *Knowledge Management Research & Practice*, vol. 16, no. 4, pp. 488–497, 2018. [Online]. Available: <https://doi.org/10.1080/14778238.2018.1475848>
- [2] R. Burkovskis, "Efficiency of Freight Forwarder's Participation in the Process of Transportation," *Transport*, vol. 23, no. 3, pp. 208–213, 2008. [Online]. Available: <https://doi.org/10.3846/1648-4142.2008.23.208-213>
- [3] J.-A. Moscoso-López, I. T. Turias, M. Come, J. Ruiz-Aguilar, and M. Cerbán, "Short-Term Forecasting of Intermodal Freight Using ANNs and SVR: Case of the Port of Algeciras Bay," *Transportation Research Procedia*, vol. 18, pp. 108–114, 2016. [Online]. Available: <https://doi.org/10.1016/j.trpro.2016.12.015>
- [4] A. Balster, O. Hansen, H. Friedrich, and A. Ludwig, "An ETA Prediction Model for Intermodal Transport Networks Based on Machine Learning," *Business & Information Systems Engineering*, vol. 62, no. 5, pp. 403–416, 2020. [Online]. Available: <https://doi.org/10.1007/s12599-020-00653-0>
- [5] S. Wickramanayake and H. D. Bandara, "Fuel Consumption Prediction of Fleet Vehicles Using Machine Learning: A Comparative Study," in *2016 Moratuwa Engineering Research Conference, MERCon 2016*, 2016, pp. 90–95. [Online]. Available: <https://doi.org/10.1109/MERCon.2016.7480121>
- [6] M. A. Hamed, M. H. Khafagy, and R. M. Badry, "Fuel Consumption Prediction Model Using Machine Learning," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 11, 2021. [Online]. Available: <https://doi.org/10.14569/IJACSA.2021.0121146>
- [7] K. Tsolaki, T. Vafeiadis, A. Nizamis, D. Ioannidis, and D. Tzovaras, "Utilizing Machine Learning on Freight Transportation and Logistics Applications: A Review," *ICT Express*, 2022. [Online]. Available: <https://doi.org/10.1016/j.ict.2022.02.001>
- [8] Y. Konishi, S.-i. Mun, Y. Nishiyama, and J. E. Sung, *Determinants of Transport Costs for Inter-regional Trade*. Research Institute of Economy, Trade and Industry, 2012.
- [9] B. Kordnejad, "Intermodal Transport Cost Model and Intermodal Distribution in Urban Freight," *Procedia – Social and Behavioral Sciences*, vol. 125, pp. 358–372, 2014. [Online]. Available: <https://doi.org/10.1016/j.sbspro.2014.01.1480>
- [10] S. Camisón-Haba and J. A. Clemente, "A Global Model for the Estimation of Transport Costs," *Economic Research – Ekonomska Istraživanja*, vol. 33, no. 1, pp. 2075–2100, 2020. [Online]. Available: <https://doi.org/10.1080/1331677X.2019.1584044>
- [11] S. Nataraj, C. Alvarez, L. Sada, A. Juan, J. Panadero, and C. Bayliss, "Applying Statistical Learning Methods for Forecasting Prices and Enhancing the Probability of Success in Logistics Tenders," *Transportation Research Procedia*, vol. 47, pp. 529–536, 2020. [Online]. Available: <https://doi.org/10.1016/j.trpro.2020.03.128>
- [12] A. Singh, A. Das, U. K. Bera, and G. M. Lee, "Prediction of Transportation Costs Using Trapezoidal Neutrosophic Fuzzy Analytic Hierarchy Process and Artificial Neural Networks," *IEEE Access*, vol. 9, pp. 103 497–103 512, 2021. [Online]. Available: <https://doi.org/10.1109/ACCESS.2021.3098657>
- [13] A. Janusz and D. Ślęzak, "KnowledgePit Meets BrightBox: A Step Toward Insightful Investigation of the Results of Data Science Competitions," in *Proceedings of the 2022 Federated Conference on Computer Science and Intelligence Systems, Sofia, Bulgaria, September 4-7, 2022*, ser. Annals of Computer Science and Information Systems, M. Ganzha, M. Paprzycki, and D. Ślęzak, Eds., vol. 30, 2022.
- [14] A. Janusz, T. Tajmayer, M. Świechowski, Ł. Grad, J. Puczniewski, and D. Ślęzak, "Toward an Intelligent HS Deck Advisor: Lessons Learned from AAI'A'18 Data Mining Competition," in *Proceedings of the 2018 Federated Conference on Computer Science and Information Systems, FedCSIS 2018, Poznań, Poland, September 9-12, 2018*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 15, 2018, pp. 189–192. [Online]. Available: <https://doi.org/10.15439/2018F386>
- [15] A. Janusz, M. Przyborowski, P. Biczysk, and D. Ślęzak, "Network Device Workload Prediction: A Data Mining Challenge at Knowledge Pit," in *Proceedings of the 2020 Federated Conference on Computer Science and Information Systems, FedCSIS 2020, Sofia, Bulgaria, September 6-9, 2020*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 21, 2020, pp. 77–80. [Online]. Available: <https://doi.org/10.15439/2020F159>
- [16] D. Ślęzak, M. Grzegorowski, A. Janusz, M. Kozielski, S. H. Nguyen, M. Sikora, S. Stawicki, and Ł. Wróbel, "A Framework for Learning and Embedding Multi-Sensor Forecasting Models into a Decision Support System: A Case Study of Methane Concentration in Coal Mines," *Information Sciences*, vol. 451-452, pp. 112–133, 2018. [Online]. Available: <https://doi.org/10.1016/j.ins.2018.04.026>
- [17] H.-M. Wong, X. Chen, H.-H. Tam, J. Lin, S. Zhang, S. Yan, X. Li, and K.-C. Wong, "Feature Selection and Feature Extraction: Highlights," in *2021 5th International Conference on Intelligent Systems, Metaheuristics & Swarm Intelligence*, ser. ISMSI 2021, New York, NY, USA, 2021, pp. 49–53. [Online]. Available: <https://doi.org/10.1145/3461598.3461606>
- [18] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16, New York, NY, USA, 2016, pp. 785–794. [Online]. Available: <https://doi.org/10.1145/2939672.2939785>
- [19] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17, Red Hook, NY, USA, 2017, pp. 3149–3157.
- [20] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "CatBoost: Unbiased Boosting with Categorical Features," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, ser. NIPS'18, Red Hook, NY, USA, 2018, pp. 6639–6649.

Considering various aspects of models' quality in the ML pipeline - application in the logistics sector

Eyad Kannout*, Michał Grodzki[†] and Marek Grzegorowski[‡]

Institute of Informatics, University of Warsaw
Banacha 2, Warsaw, Poland

Email: *eyad.kannout@mimuw.edu.pl, [†]m.grodzki@students.mimuw.edu.pl, [‡]m.grzegorowski@mimuw.edu.pl

Abstract—The industrial machine learning applications today involve developing and deploying MLOps pipelines to ensure the versatile quality of forecasting models over an extended period, simultaneously assuring the model's accuracy, stability, short training time, and resilience. In this study, we present the ML pipeline conforming to all the abovementioned aspects of models' quality formulated as a constrained multi-objective optimization problem. We also provide the reference implementation on state-of-the-art methods for data preprocessing, feature extraction, dimensionality reduction, feature and instance selection, model fitting, and ensemble blending. The experimental study on the real data set from the logistics industry confirmed the qualities of the proposed approach, as the successful participation in an international data competition did.

Index Terms—XGBoost, Dimensionality reduction, Ensemble blending, Feature selection, Feature extraction, MOO, MCDA, Logistics

I. INTRODUCTION

MACHINE learning (ML) algorithms are widely used within decision-support [1] or recommender systems [2] in many branches of the industry, like fast-moving consumer goods (FMCG) [3], e-commerce [4], logistics [5], or even hard-coal mining [6]. However, as ML models continue to run in production environments for an extended period, new expectations and concerns have also arisen. Some time ago, data scientists were expected to deliver a fine-tuned model, today, attention is paid to building ML operations (MLOps) pipelines responsible for continuous monitoring and ensuring the quality of the developed models during their functioning [7], [8]. It is also worth paying attention to the ongoing shift in the quality assurance of models' predictions, which are no longer limited to optimizing a single measure, such as accuracy or root mean square error (RMSE). It should cover more models' characteristics like stability [9], resilience [10], or interpretability [11].

The fundamental task of data analysis is to represent the data accordingly to the investigated problem. The selection of appropriate criteria for assessing the quality of generated predictions is no less important. The choice of such measure primarily depends on the nature of the problem, e.g., there are different ones for classification and regression. The quality measure we choose in the model optimization process will have a crucial impact on its performance. Once decided, we

can rely on AutoML meta-learning to ignite a versatile exploration of several learning algorithms meanwhile optimizing their parameters, which, however, is very costly and time-consuming [12]. Whereas, over-optimizing a single measure in many applications is simply unnecessary. In particular, further tuning a model of sufficient quality may lead to overfitting, increase complexity, and reduce interpretability, not to mention the longer learning time and the increased cost of computing resources. Furthermore, in many cases, optimizing a single quality measure is insufficient. Reaching the optimal regression model according to the RMSE, which meanwhile is vulnerable to data deficiencies (e.g., unavailability of selected attributes), is pointless. One of the ways we may address those concerns is to refer to the multi-objective optimization (MOO) [13]. However, as the result of MOO, we do not end up with a single solution but many Pareto optimal models. Selecting the best one is still a complex and time-consuming task related to the multi-criteria decision aiding (MCDA) [14].

In response to the above expectations and challenges, let us present the ML pipeline for training forecasting models that allows optimizing not only a single quality measure, such as RMSE, accuracy, or F1-score, but also taking into account the robustness and resilience of the ensemble blended. The developed pipeline assumes that during the training procedure, the goal is not to optimize the model over days to achieve even a minimal quality improvement on a single error measure but to adhere to many business expectations possibly fast. Accordingly, we define the task as a MOO and adapt ϵ -constrained scalarization for the investigated criteria [14]. By referring to quality thresholds that correspond directly to business expectations, we could significantly limit the time of model fitting (from days to seconds), which obviously determines the lower cost of cloud computing resources [15]. Furthermore, the adopted principle of building a model on random subsets of attributes and rows allows to achieve a variety of different models within an ensemble [16], [17]. Such an approach to training set selection enables a very straightforward parallelization of the learning procedure and hence provides significant acceleration of computation [18].

Yet another material aspect of the developed solution is the proposed feature extraction mechanism. The method is composed of several steps. Firstly, we combine available data sources into one flat file and aggregate the one-to-many relations with the common *SELECT . . . GROUP BY SQL-*

Research co-funded by Polish National Science Centre (NCN) grant no. 2018/31/N/ST6/00610.

based approach to extract some generic statistics. Later, we use feature extraction methods like one-hot encoding and ordinal coding to obtain a numerical representation of the data. This way, we achieve a sparse data representation, which poses a big problem for the boosting tree algorithms by impacting the quality of their cuts on the attributes. Such a situation imposes the construction of deeper trees, making generalization difficult and leading to over-fitting. Therefore, after encoding a given feature, we apply one of the most popular dimensionality reduction methods - principal component analysis (PCA) - to use the first few components.

To show the particular qualities of our solution, we present a case study in the logistics industry for predicting costs associated with forwarding contracts. For this purpose, we used three data sets from the machine learning contest organized on the KnowledgePit.ml platform [19], which we combined together, preprocessed, and analyzed with the developed solution. In the conducted research, we assume that the acceptable level of the prediction error measured with the RMSE measure should not exceed 2.5% of the average cost of forwarding contracts in data that corresponds to RMSE of approx 0.17. We also assume the robustness threshold of 0.02, understood as the maximal acceptable difference of RMSE achieved by the model on the training and validation set during the training procedure. Furthermore, we set the resilience threshold so the constructed ensemble should consist of at least 10 models. This way, we could provide reliable forecasts even if some of the models within the ensemble became unreliable and could impair the overall prediction quality of the ensemble. Such a situation may occur in production environments, e.g., due to a software error or unavailability of the critical attributes for this model.

The main contributions of this paper are as follows:

- 1) The ML pipeline considering various aspects of models' quality formulated as a constrained multi-objective optimization problem.
- 2) The complete reference implementation of the ML pipeline providing methods for preprocessing, feature extraction, dimensionality reduction, feature and instance selection, model fitting, and ensemble blending.
- 3) The experimental study on the real data set from the logistics industry that confirmed several qualities of the proposed approach, including small prediction error (RMSE), robustness to over-fitting, fast computing time, and resilience.

The rest of the paper is organized as follows. In Section II, we review the related literature. Section III provides a complete reference for the developed ML pipeline. In Section IV, we describe in detail the experiments conducted in this study including the description of the data, experimental setup, and the results. Finally, in Section V, we draw conclusions and suggest possible future research directions.

II. RELATED WORKS

Due to the general availability and affordability of cloud services [15], and the proven effectiveness of machine learning

[5], [17], modern enterprises massively automate their processes and optimize decision-making with intelligent use of the collected data [3]. This trend is beneficial to many industries, including supply management and logistics [20]. Let us pay special attention to international freight transportation, which is related to moving goods between countries and may involve many stakeholders: shippers, carriers, forwarders, third-party logistics services, and customs of two or more countries for each movement [21]. In this context, machine learning is seen as one of the primary enablers for the dynamic development of enterprises, allowing for apt data-driven decisions, including route planning, travel time prediction, vehicle scheduling, estimated time of arrival, and foremost accurately predicting costs related to the execution of forwarding contracts [22], [23].

We can model this task as a regression of the forwarding contract costs conditioned by the attributes of orders, such as the type of order, basic characteristics of the shipped goods (e.g., dimensions, special requirements), and the expected route that a driver will have to cover. Among the ML algorithms commonly applied to solve the regression problems, we may point out eXtreme gradient boosting trees (XGBoost), deep neural networks, or support vector machines [21], [24]. Considering the industry specifics and the dynamics of changes in the business and technological environment, the developed data-driven decision-making system should promptly adapt to changes and operate reliably even in the event of data deficiencies. One of the ways to simultaneously address several potentially conflicting concerns is multi-objective optimization (MOO) [13].

Classically, MOO problems are often solved using scalarization techniques. In brief, scalarization means that the objective functions are aggregated (or reformulated as constraints), and then a single-objective problem is solved [25]. However, this method requires defining the perfect balance between objectives' importance. Another possibility to solve such a problem is to rely on Pareto front (PF) methods. For instance, the ϵ -constraint approach can obtain a set of PF solutions by keeping only one objective and subdividing the others into several segments with some thresholds. Here, we do not end up with a single solution but potentially many models, and selecting the optimal one requires further effort [14]. In the proposed framework, we refer to ϵ -constraint filtering, but instead of choosing a single model, we blend the ensemble of several solutions [17]. This way, we not only avoid the multi-criteria decision task but also introduce the additional resilience level to our solution [10].

Among the popular ensembling techniques, we may mention random forest and XGBoost. These approaches of blending tree models minimize the regression (or decision) trees' tendency of overfitting, hence, ensuring better robustness and stability [9], [24]. The stability, RMSE, and resilience can be further improved by ensuring that the trained models in the final ensemble are relatively different from each other. One way to do this is to train models on diverse subsets of objects and attributes [17]. The training set selection, complemented by parallelization of computation, can lead to better general-

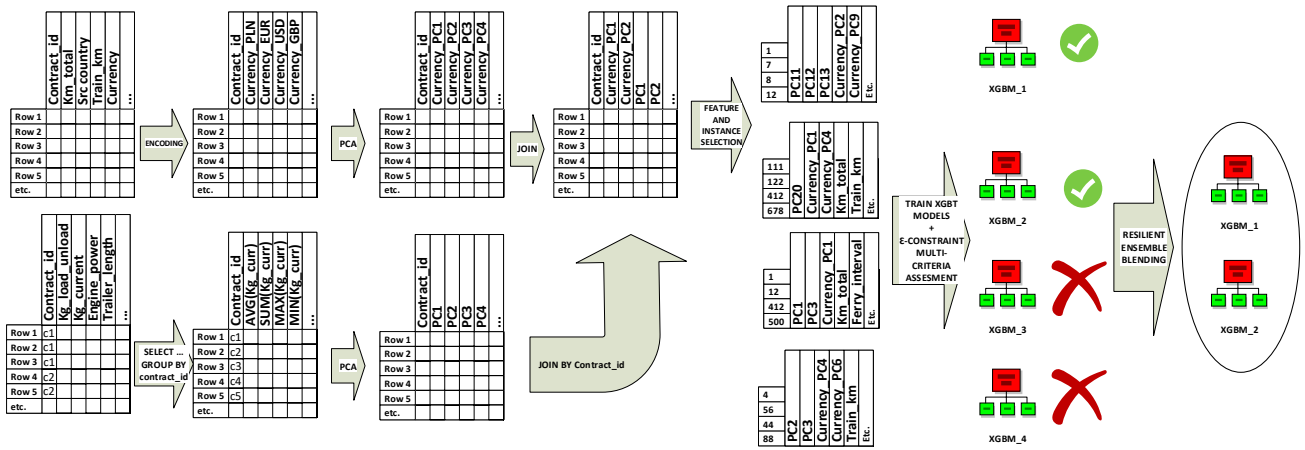


Fig. 1: Schematic ML pipeline implementation for the logistics data.

ization and minimize the overall training latency [16], [18]. It is usually essential to ingest several diverse data sources to provide adequately rich data representation with many different attributes. However, models such as XGBoost or deep neural networks operate on numerical features, whereas in most databases, we will also find some other data types.

A typical approach, in this case, would be to apply feature extraction, in particular, to encode each feature in numerical form [26]. One of the weaknesses of this approach may be the creation of a very sparse data representation related to the one-hot encoding of categorical features, which in turn can have a negative impact on tree models' performance.

Furthermore, as the number of features increases, the model training takes far more time and consumes more resources. There are many methods allowing to project or embed the data into a lower-dimensional space while retaining as much information as possible, just to mention independent component analysis, multidimensional scaling, or principal component analysis (PCA) [26]. The central idea of PCA is to reduce the dimensionality retaining as much as possible of the variation present in the data [27]. In the proposed framework, we refer to all the above-mentioned techniques. Furthermore, putting the things together, we propose the end-to-end ML pipeline adhering to the ML operations paradigm that ensures the solution's quality over time.

III. SOLUTION OVERVIEW

The developed ML pipeline consisted of several stages related to data ingestion, preprocessing, extending representation, reducing dimensionality, robust model training, and resilient ensemble blending. The first step in the developed pipeline is data ingestion and integration. Next, we perform data cleansing, encoding categorical variables to the numeric form, extracting some custom characteristics from text columns, imputing missing values, and conducting further feature extraction (FE) [18], [28].

FE addresses the problem of finding the most compact and informative data representation and is fundamental for every

ML pipeline. The importance of proper data representation was aptly identified by *Pedro Domingos*: “At the end of the day, some machine learning projects succeed, and some fail. What makes the difference? Easily the most important factor is the features used”. Using more relevant data sources and better knowledge representation may have a crucial impact on the final model quality. Therefore we plan to further extend the developed FE methods by introducing histogram-based feature engineering [29]. Among other relevant, recently reported approaches that could be in the future implemented in our ML framework, we may indicate embedding selected statistics from survival analysis or features derived from deep learning methods into data representation [5], [30].

These activities sometimes require additional effort that may depend on the data. Hence, they may not always be fully automated. For example, in the discussed case study of forwarding contracts, we ingest three data sources: *css_main*, *css_routes*, and *fuel_prices* tables (cf. Section IV-A). However, to integrate *css_routes*, we have to aggregate the data first. We execute this with the aggregating query: *SELECT AVG(.), MIN(.), MAX(.), SUM(.), COUNT(.)... GROUP BY id_contract* for each interesting variable in the table. Considering that the attributes obtained in this way are not intended for financial settlements but to feed the machine learning procedure, a vital extension of our approach would be to rely on approximated results of SQL instead of exact ones [31]. Approximate query engines can generate summaries of Big Data sets much faster with only a slight loss in precision, which may be negligible in model generalization [32], [33].

Instead of using all the SQL results for fitting the predictive model, the outcomes of the aggregation queries are transformed with PCA, and several first components are integrated into the main data. The schematic view of this process, related to the discussed case study, you may find on the left part of Figure 1. In general, whenever the encoding of categorical features into numerics significantly increases data sparsity, the derived variables are encoded with PCA, and a few first

components are kept. At the same time, the rest may be omitted. To provide an example, in Figure 2, we present the variance explained by the first 10 principal components (PC) of one-hot encoded `first_load_country` attribute.

The central part of the pipeline is selecting the training set, i.e., features and instances, to achieve the best performance and robustness of the models. In the developed approach, we iteratively draw a subset of attributes and instances as a ratio of the original data controlled by two thresholds: ω_r and ω_c (cf. Algorithm 1). In the next step, we train the selected predictive model (e.g., XGBoost or LightGBM) [34]. Since we fit the predictive model to significantly smaller data chunks, we naturally minimize the time of this process. The selection of training subsets is random and independent from each other. Hence, this process may also be easily parallelized, e.g., by drawing several subset candidates simultaneously and training the models in parallel. We also see a potential to extend our framework with the heuristic search over the subsets of attributes and features to reach the optimal quality (i.e., minimize reported error) faster [35]–[37], and to introduce more advanced feature selection techniques [26]. In this context, granular feature selection techniques could be a perfect fit [38], including r-C-reducts, bi-reducts [39], or reviving the concept of dynamic reducts [40]. Besides more advanced feature selection algorithms, instance selection has space for improvement as well. Ordering the records by date and considering only the newest instances while drawing the subsets for training data is one of many possible ideas. Combining those in an ensemble could yield interesting results.

Data:

- *dataTrain*, *dataTest* - training and test data
- θ - acceptable RMSE threshold
- ϑ - stability threshold for RMSE
- ρ - expected resilience level
- ω_r and ω_c - instance and feature selection thresholds
- *score()*- quality measure, e.g., RMSE
- *N* - maximal number of unsuccessful attempts

Result: *ensemble of models*

```

/* Initialization */
ensemble ← ∅; k ← 0
validationSet ← dataTrain.sample
dataTrain ← dataTrain \ validationSet
while |ensemble| < ρ ∧ k < N do
  trainSet ← draw ωr rows and ωc cols from dataTrain
  model ← trainXGBT(trainSet)
  Θt ← score(model, trainSet)
  Θv ← score(model, validationSet)
  if Θt < θ ∧ Θv < θ ∧ |Θt - Θv| < ϑ then
    k ← 0
    ensemble ← ensemble ∪ {model}
  else
    k ← k + 1
end
end
return ensemble;

```

Algorithm 1: Resilient and stable ensemble blending

Each model fitted on training subsets is assessed with the ε -constrained approach to handle multiple quality criteria, as

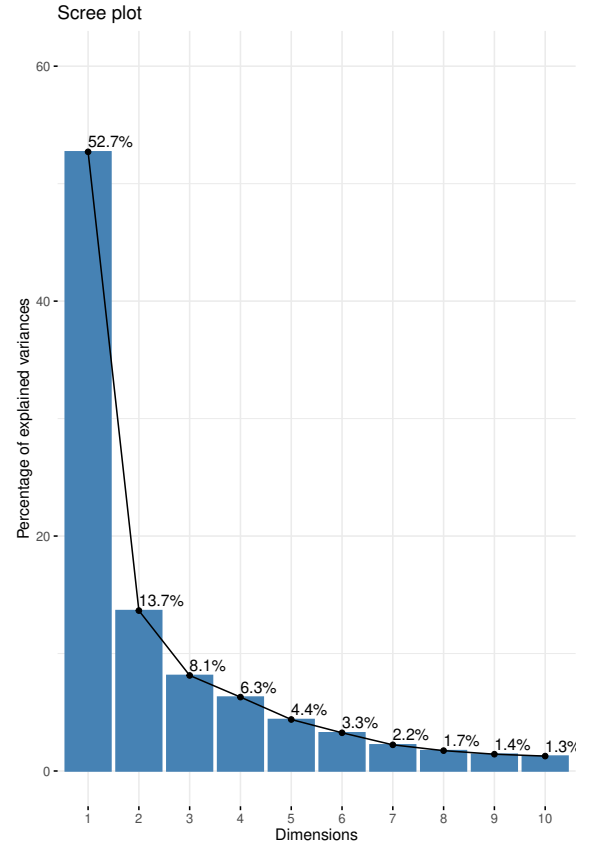


Fig. 2: Variance explained by the first ten PCA components of one-hot encoded `first_load_country` attribute.

presented in Algorithm 1. In particular, the primary objective function is to minimize an error measure, e.g., RMSE. We guarantee that each of the trained models yields an error lower than the predefined threshold θ on validation data. Furthermore, we introduce a stability threshold ϑ to avoid overfitting. We calculate it as an absolute difference between errors reported on training and validation sets in each round. The best models are blended to form an ensemble of possibly diverse models. The additional parameter ρ determines the number of models within the ensemble to provide a certain resilience level in case some of the models were put out of action.

In the future, we plan to extend a multicriteria evaluation with a specific approach to assure difference between the models explicitly. One of the possible solutions could be measuring the distances between the reported scores on a validation set. Alternatively, we may assure the feature importance rankings reported by models are possibly dissimilar (cf. Figures 4). Another approach to construct ensembles of possibly diverse models could be achieved by referring to r-C-reducts on nonoverlapping feature subsets or by ensuring constraints between attribute sets [10], [41]. A combination of the above techniques would also be an exciting future research direction, primarily since many attributes are somehow related,

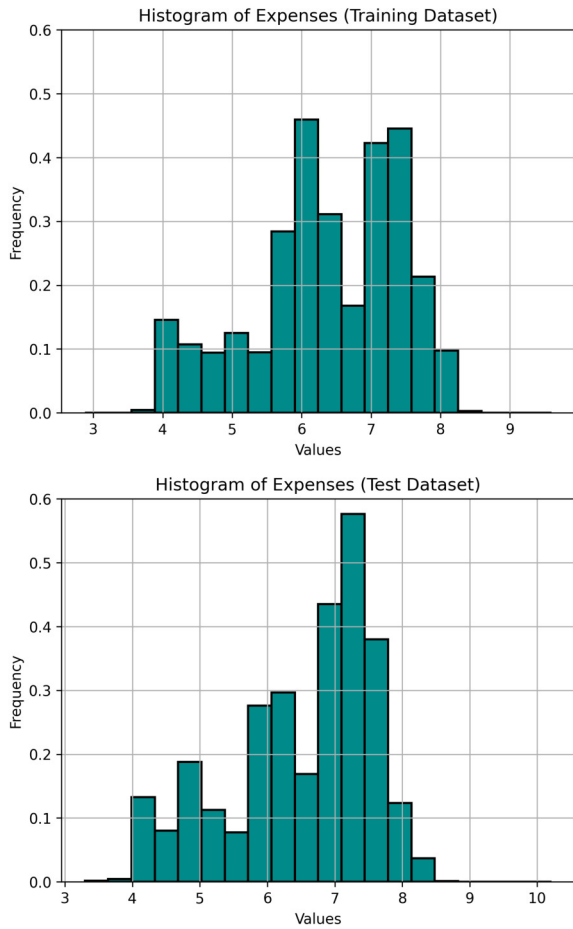


Fig. 3: Histogram of the target variable (expenses) in the experimental data (on top) and the FedCSIS competition’s final dataset (bottom).

e.g., principal components generated from the same original features or all the columns in databases referring to the same source of knowledge. Combining these techniques with state-of-the-art instance selection and training set selection methods would also be of great importance [42].

Both in wrapper search and model training, we focused mainly on XGBoost [43]. However, we also complemented it with an evaluation of LightGBM [34]. We conducted grid search model parameter tuning for best performing data subsets in the feature selection stage. This part was conducted iteratively and was alternated with the wrapper-based feature selection. Such an iterative approach allowed us to, over time, fine-tune the wrappers’ configuration. In the future, we plan to continue experiments with more advanced techniques of searching the hyper-parameter space in order to optimize the model faster and more efficiently [44].

IV. EXPERIMENTS

A. Data

The data sets contain 6 years of history of orders appearing on the transport exchange, along with details such as the type of order, basic characteristics of the shipped goods (e.g., dimensions, special requirements), as well as the expected route that a driver will have to cover (cf. Table I). In particular, the training data consist of two tables: `css_main_training.csv` and `css_routes_training.csv`, and the additional data table containing historical wholesale fuel prices for the period of training and test data. The first file (i.e., `css_main_training.csv`) contains basic information about the contracts, and the second one (i.e., `css_routes_training.csv`) describes the main sections of the planned routes associated with each contract. In both tables, the first column (i.e., `id_contract`) contains identifiers that allow matching records from `css_main_training.csv` and `css_routes_training.csv` files. Additionally, the second column in the `css_main_training.csv` file (i.e., `expenses`) contains information about the prediction target. Values in this column are available only for the training data.

B. Task and experimental setup

In our study, we investigated the task of predicting the costs related to the execution of forwarding contracts, which was defined within the 8th data mining competition organized online on the KnowledgePit platform in association with the Federated Conference on Computer Science and Information Systems (FedCSIS’22). The task is to design an accurate method for regression of costs associated with forwarding contracts [45], based on contract data and planned routes (cf. Section IV-A). The quality of predictions was evaluated with the RMSE measure. The experiments were also planned to validate the relation between training speed and the size of training data, controlled with the ω_r and ω_c parameters.

Many threats may impact the models’ performance or impede their operations, including missing data or software errors. Therefore, the experimental setup was designed to also evaluate the resilience of the final solution, understood as the RMSE error achieved when some models within the ensemble cannot be applied (e.g., due to a software error or missing data attributes). However, up to our knowledge, there is no established methodology allowing us to assess the resilience of the ML models. One of the approaches could be to randomly delete subsets of test data, e.g., by dropping particular columns. It is, however, not straightforward how to implement this kind of test. Shall we randomly drop a certain percent (e.g., 5% or 10%) or number (3 or 5) of all columns? For data sets containing 20000 attributes, such operation may have minimal impact on predictive models [6], [17]. Or shall we drop the model’s most important feature(s)? Different approaches would be preferable for other modeling techniques. Consider a random forest that relies on many redundant weak tree models, XGBoost where particular predictors are boosting the formerly selected ones, or a single tree that depends on just a few attributes with one surrogate or verifying cut per split

TABLE I: Competition data description

Data type	Example columns	Description	Processing
Categorical data	id_payer, id_currency, direction, load_size_type, service_type, contract type, first_load_country, last_unload_country	Information about transport type, start and destination country, contract currency	one-hot encoding and PCA
GPS data	first_load_lat, first_load_lon, last_unload_lat, last_unload_lon, route_start_lat, route_start_lon, route_end_lat, route_end_lon	GPS coordinates of transport start and end points	NA
Numerical data	km_total, km_empty, km_nonempty, prim_ferry_line, ferry_duration, ferry_intervals, max_weight	Information about total distances to be covered with each mode of transport, weight of the payload, current fuel prices, aggregated information about the planned route	Aggregation
Binary data	refrigerated, if_empty	Additional information about the payload, for example if it was refrigerated	NA
Date data	route_start_datetime, route_end_datetime	When the service was executed	NA

[46]. In our case, we decided to implement a straightforward approach that may be dedicated to the ensembling techniques. Namely, we were dropping randomly selected models from the ensemble to measure the impact of such an operation on the RMSE.

In the conducted experiments, we used the features extraction, dimensionality reduction, model training, and resilient ensemble blending method, as described in Section III. As the base model, we used XGBoost a [24]. We optimized the model parameters only once on the selected subset of data with the grid search procedure, and since it was not the major point of our research, we kept those parameters later unchanged. The ω_r and ω_c parameters were set to 0.5, meaning that each of the xgbt models within the ensemble was trained on a random subset of 50% features and 50% instances from the training set. The expected model quality threshold θ was set to 0.17, which corresponds to 2.5% of the median expenses in data (cf. Figures 3). The robustness threshold ϑ was 0.02, and the resilience threshold ρ was 10. To visualize the results, we use box plots - a standard way of displaying data distribution by encoding their five key characteristics: minimum, first quartile (Q1), median, third quartile (Q3), and maximum. For the purpose of evaluation, we split the data into three sets: training, validation, and test. The first two constituted a part of the training procedure. The last was used only for evaluation. Furthermore, we present the results achieved by our method within the "FedCSIS 2022 Challenge: Predicting the Costs of Forwarding Contracts" on the preliminary and final competition data.

C. Results

Figure 5 shows how the values of RMSE are spread out for ensemble models on the training, validation, and test data. The results show that RMSE for training, validation, and test dataset are very close to each other avoiding model overfitting. This confirmed that the ensemble yields not only satisfactory quality but also guarantees high robustness and stability, which is especially visible between validation and test sets (cf. Figure 5). This confirms the effectiveness of the multicriterial evaluation, such as acceptable RMSE threshold θ and stability threshold for RMSE ϑ , which are applied while selecting the ensemble models. When we compare the

results achieved by our method within our experimental setup with the results achieved during the FedCSIS 2022 Challenge: Predicting the Costs of Forwarding Contracts, we may notice it yielded very similar results on both preliminary and final data sets, 0.165 and 0.161, respectively. The results confirm the predictability, repeatability, and stability of the proposed method.

Next, we evaluate the resilience of our solution. In this experiment, we randomly deleted 1, 3, 5, 7, and 9 models from the ensemble to simulate the scenario when some models would be unavailable. Then, we calculated RMSE against the test dataset. We repeated the procedure 10 times and plotted the results in Figure 6. The results show that RMSE is slightly decreasing when deleting some models from the ensemble. However, the largest impact was spotted when deleting the majority of the models (9 out of 10). In fact, this leads us to investigate the most important features which are very correlated or have a high impact on the target variable. Therefore, we calculated the F-score based on the Information Gain (IG) measure. It is worth noting that IG in the decision/regression tree-based models is a measure of how much information a feature provides about the target feature.

Figure 4 shows the relative importance of features for two selected XGBoost models from the ensemble. The values on the x-axis show the average gain for the top ten features across all splits where those features were used. Observably, the proposed training set selection procedure allowed us to train several significantly different models that rely on different features, which leveraged ensemble smoothing and enabled the high resilience of the final solution. Considering the importance of features for 10 XGBoost models constituting the final solution (cf. Fig 4), we may notice that only seven features were relatively impactful (i.e., F-score > 100) for more than one model, namely: diff_start_end_days, diff_start_end_weeks, direction_PC1, km_nonempty, km_total, last_unload_country_PC2, last_unload_country_PC5.

For some of the potential threats, like data deficiencies, we may notice that several attributes were derived from the same (or similar) sources of information, e.g., features derived from the start and end dates or principal components of one-hot-encoded last_unload_country. Thus, in such a case,

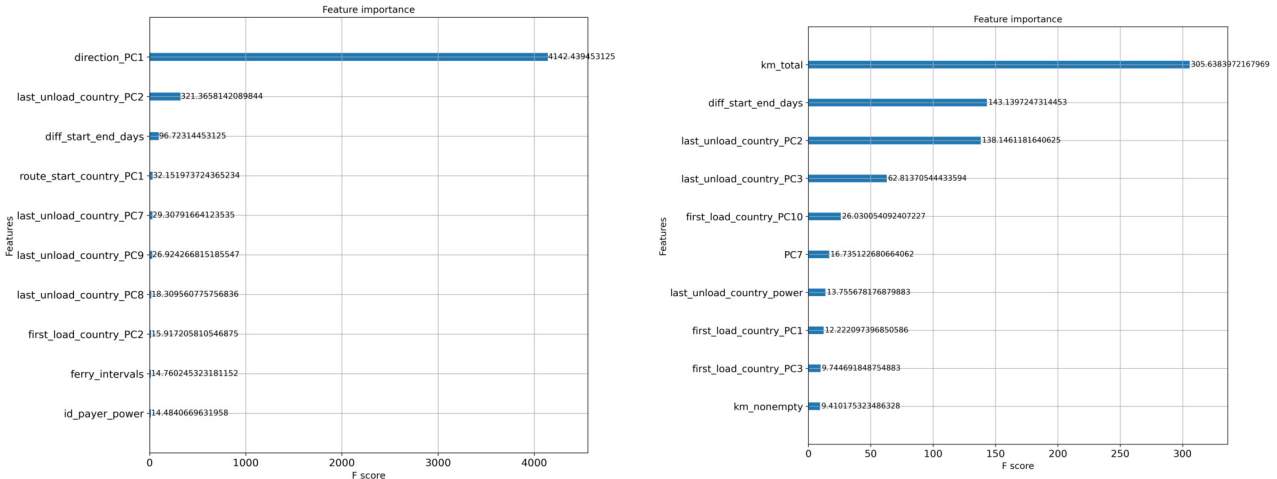


Fig. 4: Feature importance reported by two XGBoost models trained on different subsets of features and instances.

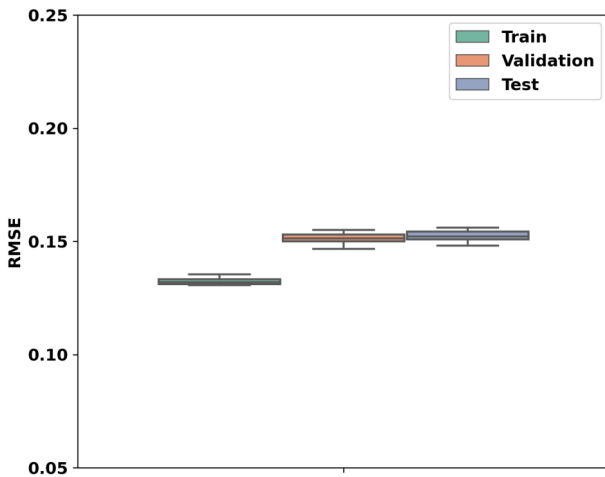


Fig. 5: RMSE comparison for ensemble models.

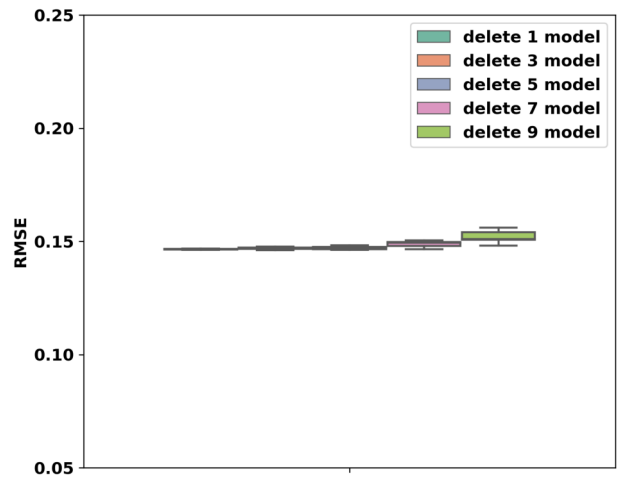


Fig. 6: Resilience comparison for ensemble models.

applying a more formal methodology for resilience assessment would be advisable and may be considered a valid future research direction. Another valuable extension of the proposed framework would be to include information from experts in the machine learning process to ensure that the features which, according to the experts, have high importance are selected in at least some of the ensemble components [26], [47].

Finally, we focus on measuring the speed and cost-effectiveness of the solution. This factor has recently gained a lot of attention because, in many practical applications of ML, like recommender or threat detection systems, the predictions are highly influenced by the most recent data. Thus, the model must be continually retrained to consider the most recent information, and it is very important to make a balance between speed and accuracy. In our proposed solution, this can be achieved by training data sets of randomly selected chunks of data. Figure 7 shows the time taken for building the XGBoost model using 100%, 75%, 50%, and 25% of

data rows (cf. ω_T parameter) in the training data set, with the unchanged model hyper-parameters. This experiment was repeated 10 times, considering different (randomly chosen) data rows in each run. Furthermore, in Figure 8, we also plot RMSE each XGBoost model would achieve in FedCSIS'22 data competition. We may notice that depending on the data subset, the final model quality varies and slightly decreases along with the declining size of training data chunks.

V. SUMMARY

The industrial machine learning applications today involve the development and deployment of MLOps pipelines, which consist of automated activities that were once manually performed by data analysts, including data ingestion and pre-processing, feature extraction and selection, model fitting, etc. These solutions are designed to ensure the quality of predictions during the production use of forecasting models, which may be months or even years. Quality assurance in such a long period requires the development of a repeatable

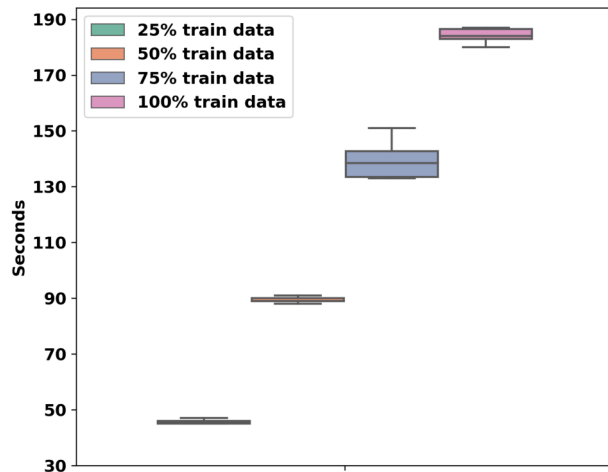


Fig. 7: Training speed comparison.

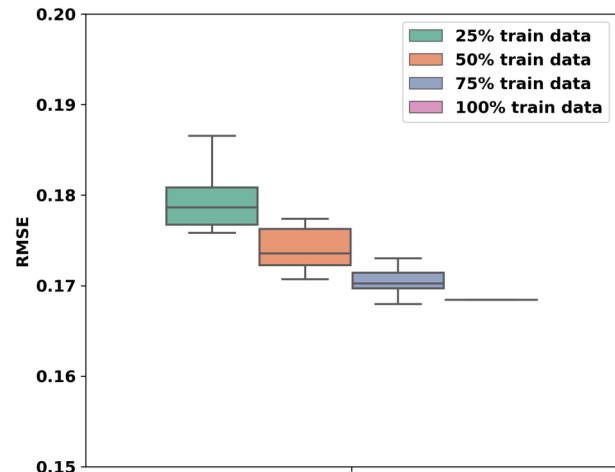


Fig. 8: RMSE comparison before/after deleting data rows.

learning procedure to (re)train the models in case of shifts or drifts in data. Furthermore, apart from confirming the model's accuracy, it is expected to assure many quality criteria, such as stability, resilience, and low computational cost, in a fast and reliable way.

In this study, we present the ML pipeline that considers several qualities of the models in a multicriterial manner. Besides assuring RMSE optimization, our solution also ensures robustness, resilience, and instant (re)training time. The developed pipeline consists of several states, including pre-processing, feature extraction, dimensionality reduction, robust model training, and resilient ensemble blending. In this paper, we also elaborate on several promising future research directions, including applying more advanced features and instances selection techniques, incorporating experts' knowledge into the machine learning processes, ensuring the ensemble diversity more explicitly, or providing a formal methodology to assess the resilience of the predictive models.

We confirmed the qualities of our pipeline with the versatile experimentation on the real data from international freight forwarders and by participating in an international data mining competition organized along to FedSCSIS'22 conference. The achieved RMSE is comparable to the best and most complex models reported by 135 teams from 24 countries in the FedCSIS contest, meanwhile conforming to more requirements. We may conclude that the proposed solution provides high-quality results with excellent resilience and stability, and the models are developed within seconds of training on low-cost compute resources. In the future, we plan to augment the developed framework with the discussed extensions and subject it to in-depth experimental analysis on a more significant number of real data sets from various fields, including the mining industry [17], [18], fire service [28], FMCG [3], cloud resource management [15], and for predicting escalations in customer support [48]. We believe that the developed approach will be equally effective in all those applications.

REFERENCES

- [1] E. Zdravetski, P. Lameski, C. Apanowicz, and D. Ślęzak, "From big data to business analytics: The case study of churn prediction," *Appl. Soft Comput.*, vol. 90, p. 106164, 2020. doi: 10.1016/j.asoc.2020.106164
- [2] E. Kannout, "Context Clustering-based Recommender Systems," in *2020 15th Conference on Computer Science and Information Systems (FedCSIS)*, 2020. doi: 10.15439/2020F54 pp. 85–91.
- [3] M. Grzegorowski, A. Janusz, S. Lazewski, M. Swiechowski, and M. Jankowska, "Prescriptive analytics for optimization of fmccg delivery plans," in *Proceedings of IPMU'22*, 2022.
- [4] Y. Li, Y. Yang, K. Zhu, and J. Zhang, "Clothing sale forecasting by a composite gru–prophet model with an attention mechanism," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 12, pp. 8335–8344, 2021. doi: 10.1109/TII.2021.3057922
- [5] M. Grzegorowski, J. Litwin, M. Wnuk, M. Pabis, and L. Marcinowski, "Survival-based feature extraction - application in supply management for dispersed vending machines," *IEEE Transactions on Industrial Informatics*, 2022. doi: 10.1109/TII.2022.3178547
- [6] D. Ślęzak, M. Grzegorowski, A. Janusz, M. Kozielski, S. H. Nguyen, M. Sikora, S. Stawicki, and L. Wrobel, "A Framework for Learning and Embedding Multi-Sensor Forecasting Models into a Decision Support System: A Case Study of Methane Concentration in Coal Mines," *Information Sciences*, vol. 451–452, pp. 112–133, 2018.
- [7] C. Renggli, L. Rimanic, N. M. Gürel, B. Karlas, W. Wu, and C. Zhang, "A Data Quality-Driven View of MLOps," *CoRR*, vol. abs/2102.07750, 2021. [Online]. Available: <https://arxiv.org/abs/2102.07750>
- [8] Y. Zhou, Y. Yu, and B. Ding, "Towards MLOps: A Case Study of ML Pipeline Platform," in *2020 International Conference on Artificial Intelligence and Computer Engineering (ICAICE)*, 2020. doi: 10.1109/ICAICE51518.2020.00102 pp. 494–500.
- [9] A. Subbaswamy, R. Adams, and S. Saria, "Evaluating Model Robustness and Stability to Dataset Shift," in *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, A. Banerjee and K. Fukumizu, Eds., vol. 130. PMLR, 13–15 Apr 2021, pp. 2611–2619. [Online]. Available: <https://proceedings.mlr.press/v130/subbaswamy21a.html>
- [10] M. Grzegorowski and D. Ślęzak, "On resilient feature selection: Computational foundations of r-C-reducts," *Inf. Sci.*, vol. 499, pp. 25–44, 2019. doi: 10.1016/j.ins.2019.05.041
- [11] C. Rudin, "Please Stop Explaining Black Box Models for High Stakes Decisions," *CoRR*, vol. abs/1811.10154, 2018.
- [12] X. He, K. Zhao, and X. Chu, "Automl: A survey of the state-of-the-art," *Knowl. Based Syst.*, vol. 212, p. 106622, 2021. doi: 10.1016/j.knsys.2020.106622
- [13] J. Blank and K. Deb, "Pymoo: Multi-Objective Optimization in Python," *IEEE Access*, vol. 8, pp. 89 497–89 509, 2020. doi: 10.1109/ACCESS.2020.2990567

- [14] H. M. Ridha, C. Gomes, H. Hizam, M. Ahmadipour, A. A. Heidari, and H. Chen, "Multi-objective optimization and multi-criteria decision-making methods for optimal design of standalone photovoltaic system: A comprehensive review," *Renewable and Sustainable Energy Reviews*, vol. 135, p. 110202, 2021. doi: 10.1016/j.rser.2020.110202
- [15] M. Grzegorowski, E. Zdravevski, A. Janusz, P. Lameski, C. Apanowicz, and D. Ślęzak, "Cost Optimization for Big Data Workloads Based on Dynamic Scheduling and Cluster-Size Tuning," *Big Data Research*, vol. 25, p. 100203, 2021. doi: 10.1016/j.bdr.2021.100203
- [16] N. Verbiest, J. Derrac, C. Cornelis, S. García, and F. Herrera, "Evolutionary wrapper approaches for training set selection as preprocessing mechanism for support vector machines: Experimental evaluation and support vector analysis," *Applied Soft Computing*, vol. 38, pp. 10–22, 2016. doi: 10.1016/j.asoc.2015.09.006
- [17] A. Janusz, M. Grzegorowski, M. Michalak, Ł. Wróbel, M. Sikora, and D. Ślęzak, "Predicting Seismic Events in Coal Mines Based on Underground Sensor Measurements," *Engineering Applications of Artificial Intelligence*, vol. 64, pp. 83–94, 2017.
- [18] M. Grzegorowski, "Massively Parallel Feature Extraction Framework Application in Predicting Dangerous Seismic Events," in *Proceedings of the 2016 Federated Conference on Computer Science and Information Systems, FedCSIS 2016, Gdansk, Poland, September 11-14, 2016*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 8. IEEE, 2016. doi: 10.15439/2016F90 pp. 225–229.
- [19] A. Janusz, D. Ślęzak, S. Stawicki, and M. Rosiak, "Knowledge Pit - A Data Challenge Platform," in *Proceedings of the 24th International Workshop on Concurrency, Specification and Programming, Rzeszow, Poland, September 28-30, 2015*, ser. CEUR Workshop Proceedings, Z. Suraj and L. Czaja, Eds., vol. 1492. CEUR-WS.org, 2015, pp. 191–195.
- [20] G. F. Frederico, "From Supply Chain 4.0 to Supply Chain 5.0: Findings from a Systematic Literature Review and Research Directions," *Logistics*, vol. 5, no. 3, 2021. doi: 10.3390/logistics5030049
- [21] L. Barua, B. Zou, and Y. Zhou, "Machine learning for international freight transportation management: A comprehensive review," *Research in Transportation Business & Management*, vol. 34, p. 100453, 2020. doi: 10.1016/j.rtbm.2020.100453 Data analytics for international transportation management.
- [22] N. Servos, X. Liu, M. Teucke, and M. Freitag, "Travel Time Prediction in a Multimodal Freight Transport Relation Using Machine Learning Algorithms," *Logistics*, vol. 4, no. 1, 2020. doi: 10.3390/logistics4010001
- [23] S.-H. Chung, "Applications of smart technologies in logistics and transport: A review," *Transportation Research Part E: Logistics and Transportation Review*, vol. 153, p. 102455, 2021. doi: 10.1016/j.tre.2021.102455
- [24] J. Nobre and R. F. Neves, "Combining Principal Component Analysis, Discrete Wavelet Transform and XGBoost to trade in the financial markets," *Expert Systems with Applications*, vol. 125, pp. 181–194, 2019. doi: 10.1016/j.eswa.2019.01.083
- [25] R. Kasimbeyli, Z. Kamisli Ozturk, N. Kasimbeyli, G. Dinc Yalcin, and B. İcmen Erdem, "Comparison of Some Scalarization Methods in Multiobjective Optimization," *Bull. Malays. Math. Sci. Soc.*, vol. 42, p. 1875–1905, 09 2019. doi: 10.1007/s40840-017-0579-4
- [26] M. Grzegorowski, "Selected aspects of interactive feature extraction," Ph.D. dissertation, University of Warsaw, 2021.
- [27] D. Granato, J. S. Santos, G. B. Escher, B. L. Ferreira, and R. M. Maggio, "Use of principal component analysis (PCA) and hierarchical cluster analysis (HCA) for multivariate association between bioactive compounds and functional properties in foods: A critical perspective," *Trends in Food Science & Technology*, vol. 72, pp. 83–90, 2018. doi: 10.1016/j.tifs.2017.12.006
- [28] M. Grzegorowski and S. Stawicki, "Window-based feature extraction framework for multi-sensor data: A posture recognition case study," in *2015 Federated Conference on Computer Science and Information Systems, FedCSIS 2015, Łódź, Poland, September 13-16, 2015*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 5. IEEE, 2015. doi: 10.15439/2015F425 pp. 397–405. [Online]. Available: <https://doi.org/10.15439/2015F425>
- [29] E. Zdravevski, P. Lameski, R. Mingov, A. Kulakov, and D. Gjorgievikj, "Robust histogram-based feature engineering of time series data," in *2015 Federated Conference on Computer Science and Information Systems, FedCSIS 2015, Łódź, Poland, September 13-16, 2015*, ser. Annals of Computer Science and Information Systems, M. Ganzha, L. A. Maciaszek, and M. Paprzycki, Eds., vol. 5. IEEE, 2015. doi: 10.15439/2015F420 pp. 381–388. [Online]. Available: <https://doi.org/10.15439/2015F420>
- [30] B. Petrovska, E. Zdravevski, P. Lameski, R. Corizzo, I. Stajduhar, and J. Lerga, "Deep Learning for Feature Extraction in Remote Sensing: A Case-Study of Aerial Scene Classification," *Sensors*, vol. 20, no. 14, p. 3906, 2020. doi: 10.3390/s20143906. [Online]. Available: <https://doi.org/10.3390/s20143906>
- [31] D. Ślęzak, A. Chadzynska-Krasowska, J. Holland, P. Synak, R. Glick, and M. Perkowski, "Scalable cyber-security analytics with a new summary-based approximate query engine," in *2017 IEEE International Conference on Big Data (IEEE BigData 2017), Boston, MA, USA, December 11-14, 2017*, J. Nie, Z. Obradovic, T. Suzumura, R. Ghosh, R. Nambiar, C. Wang, H. Zang, R. Baeza-Yates, X. Hu, J. Kepner, A. Cuzzocrea, J. Tang, and M. Toyoda, Eds. IEEE Computer Society, 2017. doi: 10.1109/BigData.2017.8258128 pp. 1840–1849. [Online]. Available: <https://doi.org/10.1109/BigData.2017.8258128>
- [32] D. Ślęzak, R. Glick, P. Betlinski, and P. Synak, "A new approximate query engine based on intelligent capture and fast transformations of granulated data summaries," *J. Intell. Inf. Syst.*, vol. 50, no. 2, pp. 385–414, 2018. doi: 10.1007/s10844-017-0471-6. [Online]. Available: <https://doi.org/10.1007/s10844-017-0471-6>
- [33] M. Muniswamaiah, T. Agerwala, and C. C. Tappert, "Approximate Query Processing for Big Data in Heterogeneous Databases," in *2020 IEEE International Conference on Big Data (IEEE BigData 2020), Atlanta, GA, USA, December 10-13, 2020*, X. Wu, C. Jermaine, L. Xiong, X. Hu, O. Kotevska, S. Lu, W. Xu, S. Aluru, C. Zhai, E. Al-Masri, Z. Chen, and J. Saltz, Eds. IEEE, 2020. doi: 10.1109/BigData50022.2020.9378310 pp. 5765–5767. [Online]. Available: <https://doi.org/10.1109/BigData50022.2020.9378310>
- [34] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017. [Online]. Available: <https://proceedings.neurips.cc/paper/2017/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf>
- [35] E. Wari and W. Zhu, "A survey on metaheuristics for optimization in food manufacturing industry," *Applied Soft Computing*, vol. 46, pp. 328–343, 2016. doi: 10.1016/j.asoc.2016.04.034
- [36] M. Okulewicz and J. Mandziuk, "A metaheuristic approach to solve Dynamic Vehicle Routing Problem in continuous search space," *Swarm Evol. Comput.*, vol. 48, pp. 44–61, 2019. doi: 10.1016/j.swevo.2019.03.008. [Online]. Available: <https://doi.org/10.1016/j.swevo.2019.03.008>
- [37] M. Ulinski, A. Zychowski, M. Okulewicz, M. Zaborski, and H. Kordulewski, "Generalized Self-adapting Particle Swarm Optimization Algorithm," in *Parallel Problem Solving from Nature - PPSN XV - 15th International Conference, Coimbra, Portugal, September 8-12, 2018, Proceedings, Part I*, ser. Lecture Notes in Computer Science, A. Auger, C. M. Fonseca, N. Lourenço, P. Machado, L. Paquete, and L. D. Whitley, Eds., vol. 11101. Springer, 2018. doi: 10.1007/978-3-319-99253-2_3 pp. 29–40. [Online]. Available: https://doi.org/10.1007/978-3-319-99253-2_3
- [38] M. Grzegorowski, A. Janusz, D. Ślęzak, and M. S. Szczuka, "On the Role of Feature Space Granulation in Feature Selection Processes," in *2017 IEEE International Conference on Big Data, BigData 2017, Boston, MA, USA, December 11-14, 2017*, J. Nie, Z. Obradovic, T. Suzumura, R. Ghosh, R. Nambiar, C. Wang, H. Zang, R. Baeza-Yates, X. Hu, J. Kepner, A. Cuzzocrea, J. Tang, and M. Toyoda, Eds. IEEE Computer Society, 2017. doi: 10.1109/BigData.2017.8258124 pp. 1806–1815.
- [39] S. Stawicki, D. Ślęzak, A. Janusz, and S. Widz, "Decision Bireducts and Decision Reducts – A Comparison," *International Journal of Approximate Reasoning*, vol. 84, pp. 75–109, 2017.
- [40] J. G. Bazan, A. Skowron, and P. Synak, "Dynamic Reducts as a Tool for Extracting Laws from Decisions Tables," in *Methodologies for Intelligent Systems, 8th International Symposium, ISMIS '94, Charlotte, North Carolina, USA, October 16-19, 1994, Proceedings*, ser. Lecture Notes in Computer Science, Z. W. Ras and M. Zemankova, Eds., vol. 869. Springer, 1994. doi: 10.1007/3-540-58495-1_35 pp. 346–355.
- [41] S. H. Nguyen and M. S. Szczuka, "Feature Selection in Decision Systems with Constraints," in *Rough Sets - International Joint Conference, IJCRS 2016, Santiago de Chile, Chile, October 7-11,*

- 2016, *Proceedings*, ser. Lecture Notes in Computer Science, V. Flores, F. A. C. Gomide, A. Janusz, C. Meneses, D. Miao, G. Peters, D. Ślęzak, G. Wang, R. Weber, and Y. Yao, Eds., vol. 9920, 2016. doi: 10.1007/978-3-319-47160-0_49 pp. 537–547. [Online]. Available: https://doi.org/10.1007/978-3-319-47160-0_49
- [42] K. Killamsetty, D. Sivasubramanian, G. Ramakrishnan, and R. Iyer, “Glistar: Generalization based data subset selection for efficient and robust learning,” *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 9, pp. 8110–8118, May 2021. [Online]. Available: <https://ojs.aaai.org/index.php/AAAI/article/view/16988>
- [43] N. Zhai, P. Yao, and X. Zhou, “Multivariate Time Series Forecast in Industrial Process Based on XGBoost and GRU,” in *2020 IEEE 9th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*, vol. 9, 2020. doi: 10.1109/ITAIC49862.2020.9338878 pp. 1397–1400.
- [44] Y. Wang and X. Sherry Ni, “A XGBoost risk model via feature selection and Bayesian hyper-parameter optimization,” *International Journal of Database Management Systems*, vol. 11, no. 01, p. 01–17, Feb 2019. doi: 10.5121/ijdms.2019.11101
- [45] A. Janusz, A. Jamiołkowski, and M. Okulewicz, “Predicting the Costs of Forwarding Contracts: Analysis of Data Mining Competition Results,” in *Proceedings of the 17th Conference on Computer Science and Intelligence Systems, FedCSIS 2022, Sofia, Bulgaria, September 4-7, 2022*. IEEE, 2022.
- [46] J. G. Bazan, S. Bazan-Socha, S. Buregwa-Czuma, Ł. Dydo, W. Rząsa, and A. Skowron, “A Classifier Based on a Decision Tree with Verifying Cuts,” *Fundam. Informaticae*, vol. 143, no. 1-2, pp. 1–18, 2016. doi: 10.3233/FI-2016-1300
- [47] D. Ślęzak, M. Grzegorowski, A. Janusz, and S. Stawicki, “Toward interactive attribute selection with infolattices - A position paper,” in *Rough Sets - International Joint Conference, IJCRS 2017, Olsztyn, Poland, July 3-7, 2017, Proceedings, Part II*, ser. Lecture Notes in Computer Science, L. Polkowski, Y. Yao, P. Artimjew, D. Ciucci, D. Liu, D. Ślęzak, and B. Zielosko, Eds., vol. 10314. Springer, 2017. doi: 10.1007/978-3-319-60840-2_38 pp. 526–539. [Online]. Available: https://doi.org/10.1007/978-3-319-60840-2_38
- [48] A. Janusz, G. Hao, D. Kaluza, T. Li, R. Wojciechowski, and D. Ślęzak, “Predicting escalations in customer support: Analysis of data mining challenge results,” in *2020 IEEE International Conference on Big Data (IEEE BigData 2020), Atlanta, GA, USA, December 10-13, 2020*, X. Wu, C. Jermaine, L. Xiong, X. Hu, O. Kotevska, S. Lu, W. Xu, S. Aluru, C. Zhai, E. Al-Masri, Z. Chen, and J. Saltz, Eds. IEEE, 2020. doi: 10.1109/BigData50022.2020.9378024 pp. 5519–5526. [Online]. Available: <https://doi.org/10.1109/BigData50022.2020.9378024>

Prediction of the Costs of Forwarding Contracts with Machine Learning Methods

Stanisław Kaźmierczak

Faculty of Mathematics and Information Science,
Warsaw University of Technology, Warsaw, Poland
Email: stanislaw.kazmierczak@pw.edu.pl

Abstract—This paper summarizes experiments conducted and findings related to FedCSIS 2022 Challenge that we participated in. The task was to develop a predictive model that estimated costs pertained to the execution of forwarding contracts (FC). We thoroughly analyze the dataset and present steps performed in the data preprocessing stage. Then we describe our approach to building a predictive model, which placed us eighth out of 135 teams. In the end, a wide range of ideas for further research is provided.

Index Terms—forwarding contracts, data preprocessing, machine learning

I. INTRODUCTION

DUE TO the specialization of work, the economy of production scale, and mass consumption, places where products are manufactured do not coincide with places where the demand for them is reported. Therefore, transportation is essential to bridge the gap between the buyer and the seller. It is a part of a logistic chain and plays a crucial role in attaining its primary goal – resource optimization.

Freight forwarding is an activity consisting in organizing the transport of goods. The forwarding company performs various activities for the client related to the organization of transportation, starting from adjusting the means of transportation through consulting in the field of cargo transport and ending with unloading the goods. Forwarder operates on the freight exchange, a place of information exchange between carriers and forwarding companies. Its purpose is to facilitate communication and accelerate the conclusion of transactions in the economic sector of transport.

Forwarder's work consists in searching orders on freight exchange, evaluating them, and selecting profitable ones. The ultimate goal is to sign an FC. It is an agreement whereby the freight forwarder makes a commitment for the sender or recipient of the goods to transport them to the place of delivery, not conducting the carriage himself but finding the carrier who will carry the goods. The forwarder signs a carriage contract on his behalf but for the account of the sender or recipient.

This study describes the model created to predict the costs related to the execution of FCs. Such a model aims to support freight forwarders in selecting profitable contracts.

II. RELATED LITERATURE

Increased interest in predicting the cost of transport has been observed among researchers since the late 1990s. Various means of transport, parts of the world as well as predictive

methods were analyzed. This section summarizes more significant, in our opinion, studies.

In [1], the authors investigated factors that affected transportation costs. They applied the tobit model to find that infrastructure is its most important determinant in the considered area. Other crucial factors included details of geography, administrative barriers, and the structure of the shipping industry.

The authors of [2] used regression-based methods to examine the determinants of shipping costs to the US. It was found that distance, containerization, and efficiency of a port were significant factors influencing freight. The study provided some examples of how private involvement in port management along with labor reform and reduced monopoly power led to efficiency and lower costs.

Reference [3] studied the impact of port characteristics on international maritime transport costs. It considered 16 Latin American countries and maritime trade transactions in containerizable goods. The authors employed a regression model and proved that doubling port efficiency in a pair of ports would have the same impact on international transport costs as halving the distance between them.

The authors of [4] created a microeconomic model of interregional freight transportation. They utilized an ordinary least squared regression model and showed that besides determinants of transport cost incorporated in the model, the degree of competition also played a significant role in freight charge prediction.

Reference [5] is another paper focused on the prediction of costs of maritime transport, more precisely, logistics costs in container ports. The authors applied transaction cost economics (TCE) to support and explain empirical findings. They found that the quality of port infrastructure, port services, and port connectivity are among the most important determinants of logistics costs in container ports.

In [6], the authors summarized crucial findings from previous studies related to the estimation of transport costs. Most of the approaches concentrated only on statistical analysis or employed regression-based methods.

To our knowledge, there is no study that elaborates on more sophisticated machine learning (ML) methods and considers various means of transport. We hope that our study contributes to filling this gap.

III. DATASETS

The dataset contains orders that appeared in the freight exchange and were accepted by a large Polish company. Training samples come from the period between January 2016 and November 2020, while test instances from the period between September 2020 and November 2021. It means that train orders are generally followed by test ones. There are two main reasons why the test set partially overlaps the train set: complex orders from the training part that require a long time to complete, as well as some reversed start and end times (the end time is earlier than the start time) in 452 orders. Order details such as its type, basic characteristics of the shipped goods, along with the expected route that a driver will have to cover are provided. Input columns are the same for both training and test sets except for the target variable – costs of individual orders – which is given in the case of the former one and needs to be predicted in the case of the latter.

A. Dataset description

In more detail, the training set consists of two tables: *css_main_training.csv* and *css_routes_training.csv*. The former contains fundamental information about the contracts. It has 330 055 rows and 36 columns. The latter describes the main sections of the planned routes associated with each contract. It consists of 1 189 654 rows and 60 columns. The first column of each table contains contract identifiers that allow matching records from both tables. *css_main_training.csv* contains a column with the prediction target. Analogously, the test set consists of two tables: *css_main_test.csv* and *css_routes_test.csv*. Their structure is the same as corresponding training tables, but the column with the prediction target is empty. The first table consists of 72 452 records, while the second – 325 222 records. Additionally, *fuel_prices.csv* contains wholesale prices of three different types of fuel for the period of training and test data. More details about data and the competition itself are provided in [7].

B. Data analysis

In the case of the analyzed dataset, there are three types of data: numerical, categorical (including binary), and text (only one column of this type – requirement related to the temperature). In terms of most features, we can observe significant skewness (numerical columns) or noticeable imbalance (binary columns), which is generally not positive from a machine learning perspective.

Each order in the main table is timestamped. Thus, if orders are grouped, columns may be viewed as time series. We stick to the most important column, *expenses*, which is the target variable. Fig. 1 depicts expenses from orders grouped by different time periods. Several conclusions can be drawn. First, orders that are planned to start on Friday are the most expensive, and those beginning during a weekend – the cheapest. Conversely, orders scheduled to be finished on a weekend are high-priced. Second, throughout a year, there are peaks in cost irrespective of whether we consider the beginning or end of the order. It concerns both fixed

and floating holidays. Finally, one may observe that contracts planned to be accomplished in the summer months are cheaper than those in other parts of the year.

The results obtained in standard cross-validation in which training and test instances are mixed in terms of timestamp were approximately 10% better than those registered on the competition platform in which training samples are followed by test ones. It suggests that data or/and concept drift occurs. The difference in the distribution of some features between the training and test data is not a sufficient explanation of this phenomenon. There is no statistically significant difference in the distribution of the target variable. As the last step, we applied TSNE to map training and test instances to the 2D plane. We did not observe any significant dissimilarities between both types of samples. Fig. 2 depicts the results of the algorithm. Data/concept drift needs further investigation we did not manage to perform before the end of the competition.

IV. DATA PREPROCESSING

Quality data is necessary for machine learning models to operate efficiently. In general, data preparation requires more time and effort than actual modeling [8]. In this section, we present preprocessing steps that made our data prepared to build a predictive model.

A. Main tables

- 1) All data except the following columns were loaded: *temperature*, *first_load_lat*, *first_load_lon*, *last_unload_lat*, *last_unload_lon*, *route_start_lat*, *route_start_lon*, *route_end_lat*, *route_end_lon*.
- 2) The following columns were one-hot encoded: *direction*, *id_service_type*, *contract_type*, *id_payer*, *first_load_country*, *last_unload_country*, *route_start_country*, *route_end_country*, *id_currency*, *prim_train_line*, *load_size_type*, *prim_ferry_line*.
 - a) *id_payer* was limited to 50 payers with the most numerous contracts within both train and test set.
 - b) In terms of *prim_train_line* and *prim_ferry_line*, the additional category representing missing values was created (in other one-hot encoded columns, there were no missing values).

B. Routes tables

- 1) The following columns were loaded: *id_contract*, *external_fleet*, *id_vehicle*, *id_trailer*, *if_empty*, *ferry*, *train*, *step_type*, *country_code*, *id_vehicle_model*, *id_vehicle_type*, *vehicle_type*, *vehicle_capacity_type*, *trailer_generator*, *id_trailer_model*, *id_trailer_type*, *ferry_line*, *train_line*.
- 2) The following columns were one-hot encoded: *country_code*, *id_vehicle_model*, *id_vehicle_type*, *vehicle_capacity_type*, *step_type*, *trailer_generator*, *id_trailer_model*, *id_trailer_type*, *ferry_line*, *train_line*, *vehicle_type*.
 - a) For all columns other than *step_type*, the additional category representing missing values was created

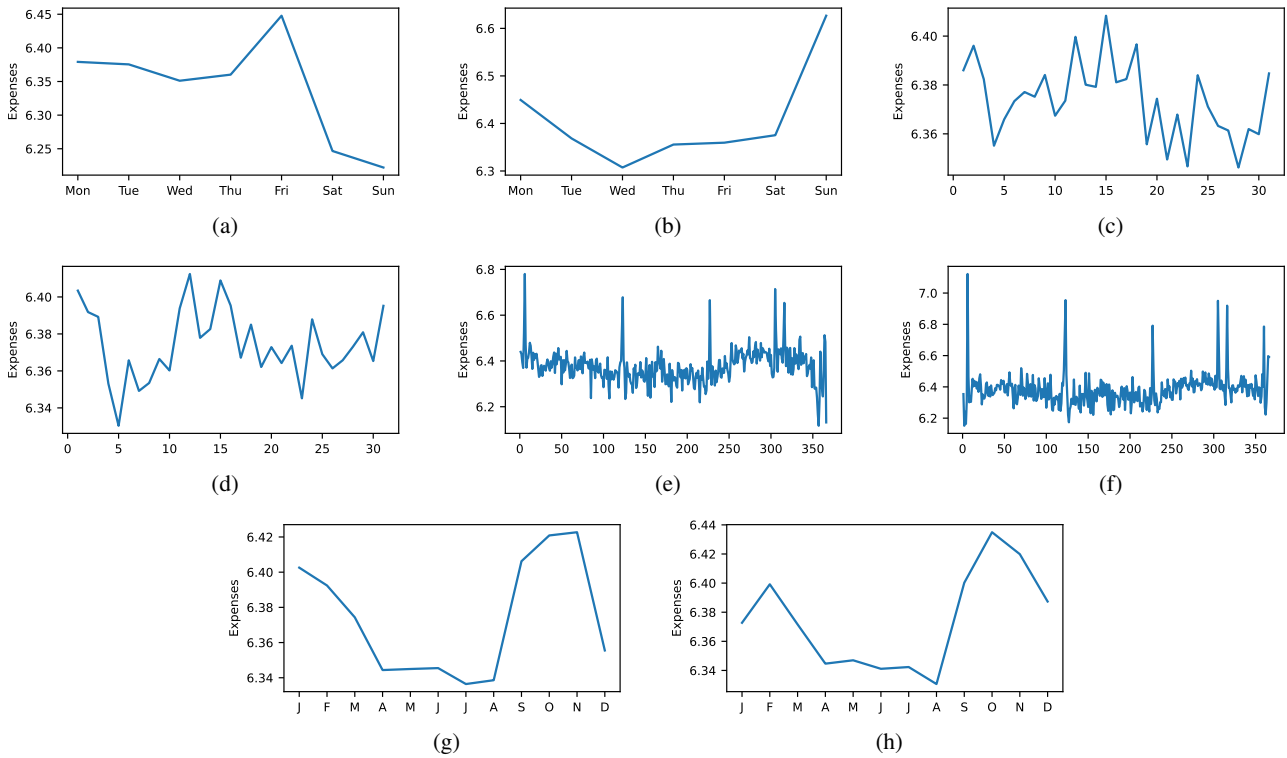


Fig. 1: Expenses (target variable) as a function of time. In the consecutive subfigures, orders are grouped by: (a) and (b) – day of a week, (c) and (d) – day of a month, (e) and (f) – day of a year, (g) and (h) – month of a year. Subfigures (a), (c), (e), and (g) relate to the planned time of the beginning of a route, subfigures (b), (d), (f), and (h) – to the planned time of the end of a route.

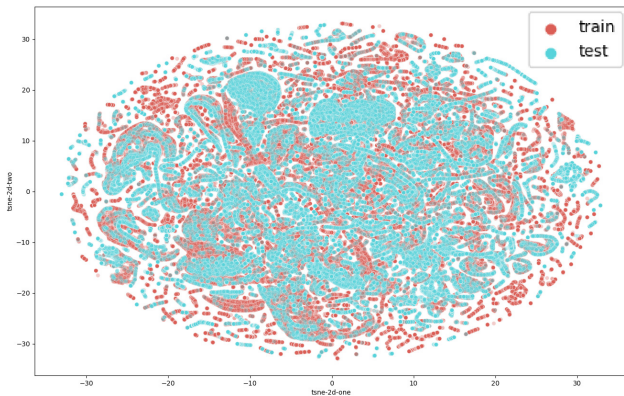


Fig. 2: TSNE applied to compare training and test instances.

(in the *step_type* column, there were no missing values).

- 3) Aggregation: instances were grouped by *id_contract*, and values from all other columns were summed for each contract. One more column reflecting the number of routes per contract was added.
- 4) Above data frames with routes were merged with the main data frames on the *id_contract* column which was

then removed.

C. Fuel prices

- 1) *disel_type2_price* column was added to the above merged data frames based on date.
 - a) *disel_type1_price* and *disel_type3_price* were not utilized since their Pearson correlation with *disel_type2_price* ranges between 0.98 — 0.99.

D. Data transformations

- 1) *ferry_intervals* was the only column with missing values. They were imputed on the basis of *ferry_duration*. If *ferry_duration* equals 0, then *ferry_intervals* is also 0. Otherwise, *ferry_intervals* is imputed with 1.
- 2) Time correction: if *route_end_datetime* was earlier than *route_start_datetime*, the values were swapped.
- 3) New features being the product of *disel_type2_price* and features related to distance (*km_empty*, *km_nonempty*, *km_total*, *km_train*, *ferry_duration*) were created. They should reflect the total cost of fuel related to particular contracts.
- 4) Additional features created:
 - a) *duration_h* — the difference expressed in hours between *route_end_datetime* and *route_start_datetime*.

- b) *start_day_of_week* and *end_day_of_week* — days of week related to *route_start_datetime* and *route_end_datetime*, respectively.
 - c) *day_of_year* — day of a year, ranges from 1 to 365 (in the case of the leap year, 1 was subtracted from all days after 29th February to be consistent with common years; it turned out that the predicted prices are highly correlated with some dates, especially related to festivals); one-hot encoded.
- 5) After all the aforementioned transformations, *route_start_datetime* and *route_end_datetime* columns were ultimately removed from the data frame.
 - 6) Eventually, the data frame contained over 1000 columns. Features were assessed using XGBoost's *feature_importance* property. For final modeling, different number of the most valuable features were left (more details are provided in Section V).

V. PREDICTION RESULTS

The task of the created models was to predict the actual costs of individual orders as accurately as possible. Such models aim to assist freight forwarders in picking beneficial contracts. The quality of algorithms is evaluated using the RMSE measure.

The whole code was written in Python 3.7. Neural networks were created in Keras 2.3.1. Gradient boosting was implemented with the *xgboost* 1.0.2 package. In terms of all other machine learning algorithms, *scikit-learn* 0.24.2 was applied. If not mentioned otherwise, hyperparameters were left at their default settings.

The best results were obtained by XGBoost built on the most valuable 200 or 500 features mentioned in Subsection IV-D. It is not a surprise in terms of the tabular data since XGBoost is the top choice on the Kaggle platform in such cases as well. In terms of hyperparameter tuning, we selected hyperparameters and their considered value range as suggested in [9] and [10]. Due to time constraints, we optimized them one by one, assuming fixed (default) values for the others. It turned out that the following values brought the best results: *subsample* – 1, *max_depth* – 6, and *eta* – 0.3.

The final solution was constituted by the averaging ensemble of three XGBoost models with *n_estimators* set to 205 and built on all, 500, and 200 most valuable features, respectively. The RMSE obtained amounted to 0.1529, which placed us eight out of 135 teams.

It is worth mentioning that many algorithms other than XGBoost, were analyzed (we submitted 108 valid solutions). Before focusing on XGBoost, we tested a wider range of algorithms – linear regression, random forest, and different neural architectures. All of them were more than 10% worse than the final solution.

VI. CONCLUSIONS AND FURTHER IDEAS

In this paper, we present our approach to building a model able to predict costs related to FC. Despite many conducted

experiments and the reasonable score achieved, we still see a lot of room for improvement.

First, concept/data drift was detected but not addressed successfully. It requires further investigation. We believe that the application of some dedicated methods (please refer to [11]) may lead to prediction enhancement.

Second, data aggregation requires more experiments. In the current approach described in subsection IV-B, values from the routes table are summed for each corresponding contract. Such a method may cause the loss of some valuable information.

Third, we believe that there is still some uncovered potential in neural networks. Neural architectures are relatively hard to tune and prone to overfitting due to their complexity. Even if they do not outperform XGBoost, they can constitute a valuable element of the ensemble model by increasing its diversity.

Next, it may be worth looking one more time at encoding categorical features with a large number of values, e.g., *id_payer*. On the one hand, we should not expand a feature space massively. On the other, we must not allow valuable information to be lost.

Last but not least, it is worth taking a closer look at the feature selection. We applied a simple approach based on XGBoost's *feature_importance* property. However, this method does not take into account feature correlation. We strongly believe that the application of some more sophisticated feature selection algorithms along with other aforementioned ideas will further boost the prediction quality.

REFERENCES

- [1] N. Limao and A. J. Venables, "Infrastructure, geographical disadvantage, transport costs, and trade," *The world bank economic review*, vol. 15, no. 3, pp. 451–479, 2001.
- [2] A. Micco and N. Pérez, "Determinants of maritime transport costs," *Inter-American Development Bank*, 2002.
- [3] G. Wilmsmeier, J. Hoffmann, and R. J. Sanchez, "The impact of port characteristics on international maritime transport costs," *Research in transportation economics*, vol. 16, pp. 117–140, 2006.
- [4] Y. Konishi, S.-i. Mun, Y. Nishiyama, and J. E. Sung, *Determinants of Transport Costs for Inter-regional Trade*. Research Inst. of Economy, Trade and Industry, 2012.
- [5] H.-s. Cho, "Determinants and effects of logistics costs in container ports: The transaction cost economics perspective," *The Asian Journal of Shipping and Logistics*, vol. 30, no. 2, pp. 193–215, 2014.
- [6] S. Camisón-Haba and J. A. Clemente, "A global model for the estimation of transport costs," *Economic research-Ekonomska istraživanja*, vol. 33, no. 1, pp. 2075–2100, 2020.
- [7] A. Janusz, A. Jamiolkowski, and M. Okulewicz, "Predicting the costs of forwarding contracts: Analysis of data mining competition results," in *Proceedings of the 17th Conference on Computer Science and Intelligence Systems, FedCSIS 2022, Sofia, Bulgaria, September 4-7, 2022*. IEEE, 2022.
- [8] S. García, J. Luengo, and F. Herrera, *Data preprocessing in data mining*. Springer, 2015, vol. 72.
- [9] A. Jain, "Complete Guide to Parameter Tuning in XGBoost with codes in Python," 2016, online; accessed 23-Jul-2022. [Online]. Available: <https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python/>
- [10] D. Martins, "XGBoost: A Complete Guide to Fine-Tune and Optimize your Model," 2021, online; accessed 23-Jul-2022. [Online]. Available: <https://towardsdatascience.com/xgboost-fine-tune-and-optimize-your-model-23d996fab663>
- [11] J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang, "Learning under concept drift: A review," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 12, pp. 2346–2363, 2018.

XGBoost meets TabNet in Predicting the Costs of Forwarding Contracts

Aleksandra Lewandowska
 Silesian University of Technology, Poland
 aleklew480@student.polsl.pl

Abstract—XGBoost and other gradient boosting frameworks are usually the default choice for solving classification and regression problems for tabular data, especially in data science competitions, as they often, combined with proper data pre-processing and feature engineering, supply high accuracy of predictions. They are also fast to learn, easy to tune, and can supply a ranking of variables, making interpretation of learned models easier. On the other hand, deep networks are the top choice for complex data, such as text, audio, or images. However, despite the many successful applications of deep networks, they are not yet prevalent on tabular ones. It may be related to difficulties in the choice of the proper architecture and its parameters. A solution to this problem may be found in recent works on deep architectures dedicated to tabular data, such as TabNet, which has recently been reported to achieve comparable or even better accuracy than XGBoost on some tabular datasets. In this paper, we compare XGBoost with TabNet in the context of the FedCSIS 2022 challenge, aimed at predicting forwarding contracts based on contract data and planned routes. The data has a typical tabular form, described by a multidimensional vector of numeric and nominal features. Of particular interest is investigating whether aggregation of predictions derived from XGBoost and TabNet could produce better results than either algorithm alone. The paper discusses the competition solution and shows some added experiments comparing XGBoost with TabNet on competition data, including incremental model re-building and parameter tuning. The experiments showed that the XGBoost and TabNet ensemble is a promising solution for building predictive models for tabular data. In the tests conducted, such an ensemble achieved a lower prediction error than each of the algorithms individually.

I. INTRODUCTION

ALMOST every company collects data on their products, services, or customers. Analyzing such data helps companies make informed business decisions to increase their profits. One example of such analysis is the task defined in the FedCSIS 2022 challenge [1], involving cost prediction of forwarding contracts based on contract details and planned routes. In that case, such cost estimation may support selecting the most profitable contracts.

Many examples of predictive analytics for supporting companies in increasing or maintaining their profits can be found [2]. Recommender systems [3], [4] may suggest to customers what products they may be interested in, thereby increasing the chances of buying a product. Predicting customer lifetime value [5] helps to select the top customers to whom more attention should be paid. Churn prediction [6], [7] can reduce customer losses.

Data collected by companies about customers, contracts, or products are often stored in database tables consisting of various attributes (nominal, numerical, date-time), thus their natural form for analytics is tabular, with mixed types of features. Over the past several years, gradient boosting frameworks have become particularly popular for analyzing such data, as combined with appropriate data pre-processing and feature engineering, they often achieve superior accuracy. In contrast, deep learning methods are the default choice for unstructured data, such as images, audio, or text. Deep networks are not often applied to tabular data, which may be due to the difficulty of selecting an appropriate architecture suitable for analyzing heterogeneous tabular formats. Therefore, several deep architectures dedicated to tabular data have been proposed over the past few years, such as TabNet [8], Net-DNF [9], Node [10], and TabTransofmer [11].

This paper attempts to apply one of those tabular-specific deep networks, namely TabNet, to forwarding contracts data in the context of the FedCSIS 2022 challenge, and compare it with XGBoost [12], one of the most popular gradient boosting frameworks. Of particular interest is investigating whether combining the two methods could produce better results than either algorithm alone. We also investigate the performance of the algorithms over time, training them in an incremental fashion. Section II describes the competition solution. In this section are described: data pre-processing steps and modelling, XGBoost hyper-parameters tuning, Tabnet hyper-parameters tuning and linear combination of both models as a result to the problem. Section III summarizes the work.

II. COMPETITION SOLUTION

A. Competition data-sets

Competition consists of five data-sets: **test data set (main table)**, **test data set (routes table)**, **training data set (main table)**, **training data set (routes table)** and **fuel prices**. Test and training data sets differ in the fact that test data set does not include the estimated cost column, which is used for evaluation purposes. The presented solution's data pre-processing process is divided into two separate steps: **routing table** and **main table** pre-processing.

B. Routing table data pre-processing

In the first step, we selected and added new features, and aggregated the data by contract id. We identified the data that was missing in a significant manner and in the process

of simplification we removed those features from further processing.

The percentage of missing values in particular columns is respectively about 54.5% and 30.6%. Therefore, the number of columns in this step has significantly dropped.

The columns describing respectively longitude and latitude were dropped as extracted information can be found in other features, like the distance between starting and ending points.

The subset of features that have not been dropped can be associated with different transport methods. The present methods are ferry, train, truck (which can be considered as conventional transportation method).

To each method we have defined added features, which measure how much has been transported over how long time or distance. Created features:

- Train (weight-distance) feature = $train_km * kg_current$
- Ferry (weight-time) feature = $ferry_duration * kg_current$
- Truck (weight-distance) feature = $km * kg_current$

As multiple records are used to define a particular transportation process, we decided to aggregate the features left. After the pre-processing process the routing table is summarized with one record per each contract.

C. Main table data pre-processing

Main table includes 36 columns, and only 8 of them has missing values. Only 3 columns include more than 50% of missing values. Therefore, those columns have been dropped.

In the first step we have extracted date and time information from **route_start_datetime** and **route_end_datetime** columns. From those columns we have extracted **year, month, day of the week, hour, the difference between start and the end** of the transportation process and converted those two columns into columns having **unix time** (which defines the number of epochs since 01-01-1970).

In the next step we put our attention towards the processing of categorical data, which can be shown on the feature **id_payer**. From the training set we selected the payers that accounted for over 1000 orders and then we applied one hot encoding technique. All not accounted payers are categorized as others. The same approach was used for: **currencies, first load country, last unload country**. Other categorical features were treated in a different manner. As the number of unique values was small enough, instead of using one hot encoding technique we have decided to assign a numerical value to each of values included within each category. To each value we assigned an integer, in our case we did not concern ourselves with the order of assigned numerical values. The numerical values used are integers from 0 to n where n depends on the number of unique values present in a particular categorical column.

Features for which assignment was applied are: **contract type, load size type** and **direction**. Then newly created features and other non-categorical features were joined with the data set created in the earlier step.

The resulting data set constitutes the input for our XGBoost and TabNet models. The same steps were applied towards test data sets in both routing table and main table data sets.

D. TabNet - training and evaluation

TabNet is an "interpretable canonical deep tabular data learning architecture." [2] The parameters that influence the training process of the model are: **optimizer** and **learning rate, hyper-parameter tuning, training process**. We used **Adam optimizer** and **exponentially decreasing learning rate of first value 2e-2**.

After choice of the optimizer and the learning rate we could turn to **hyper-parameters tuning**. For the training process we selected 4 parameters for testing [8]:

- **n_a** - "Width of the decision prediction layer. Bigger values give more ability to the model with the risk of over-fitting."
- **n_b** - "Width of the attention embedding for each mask. According to the paper $n_d = n_a$ is usually a good choice."
- **batch size** - "Number of examples per batch. Large batch sizes are recommended."
- **n_steps** - "Number of steps in the architecture (usually between 3 and 10)"

To select the most suitable parameters for our problem, we have used the technique called **Grid Search**. To use this technique, firstly we had to define the range of the parameters that we wanted to check during testing. Next, we would try all possible combinations, train the model and evaluate the performance. Selected parameter ranges are: **n_a = n_b** \in {4, 8, 18}, **n_steps** \in {3, 5, 7}, **batch size** \in {1024, 2048}, **mask type** \in {"sparsemax", "entmax"}. The table below represents a subset of results of all performed experiments. Model consistently has performed better for **mask type = "sparsemax"** and only results with such value are presented.

Width	Steps	Batch	RMSE Val	RMSE Pre
8	5	2048	0.1634	0.1719
8	5	1024	0.1608	0.1720
8	3	1024	0.1587	0.1734
8	7	1024	0.1757	0.1751
4	7	2048	0.1646	0.1752
8	3	2048	0.1618	0.1772
4	5	1024	0.1661	0.1774
8	7	2048	0.1665	0.1782
4	3	1024	0.1687	0.1779
4	3	2048	0.1667	0.1875
4	5	2048	0.1670	0.2387
4	7	1024	0.1647	0.2704

TABLE I
RESULTS OF HYPER-PARAMETER TESTING, TABNET

Width - n_a, n_b
Steps - n_steps
Batch - batch_size
RMSE Val - RMSE value, calculated on the validation set
RMSE Pre - RMSE value, calculated on the preliminary testing set

The training data set has been sorted first based on the unix epoch value of route_start_datetime column. The validation set consists of the last 10% of the sorted rows of the data set.

The sorting took place because we want our model that has been trained on the archived records to perform well on the incoming data. The main goal is to predict future values.

This exactly describes the training process that took place. In the first iteration we trained our model only on the first 10% of the data set and evaluated it on the following 10%. Therefore, only 20% has been used at this step. In the next iteration we have moved the 10% used for evaluation into the training set, on which model has been additionally trained. Then the following 10% has been used for evaluation purposes. This process was taking place incrementally, till the moment in which the model was trained on 90% of the data and evaluated on the last 10%. The RMSE value evaluated at this step is denoted as RMSE Val. However, for our model to perform well on the test data set, the model should be trained on almost all available records. For that reason, we have moved the following 8.5% from the local validation set used for evaluation and additionally trained the model, which was then evaluated on the last 1.5% of the ordered training data set. Then we considered a TabNet model trained on 98.5% of the data set as a fully trained model, which then was evaluated on the preliminary testing set, supplied by the FedCSIS 2022 competition, which is the subset of the full testing data set. The results are available in I. Hyper-parameters of the model that **performed best** are:

$n_a = n_b = 8$, $n_steps = 5$, batch size = **2048**, mask type = "sparsemax"

The results of evaluation are: local validation = **0.1634**, preliminary validation = **0.1719**, final result = **0.1713**.

The results obtained for the model with **default hyper-parameters** are: local validation = **0.1587**, preliminary validation = **0.1733**, final result = **0.1735**

We can see that the model has improved, but not in a significant manner. In comparison to the model which obtained the best results on the preliminary test data set, both models differ by the **batch size**, and **n_steps** which are respectively **1024** and **3** for the model trained with default hyper-parameters.

E. XGBoost - training and evaluation

XGBoost "is an optimized distributed gradient boosting library"[3]. XGBoost model consists of two steps: **training** and **evaluation**. Those processes are quite similar to the processes that were described in the TabNet section of this paper. The training process is similar to the TabNet training process with a subtle differences. Similarly we use **Grid Search** technique for finding the optimal hyper-parameters for our model. The hyper-parameters that were tuned are:

- **learning rate** - "Step size shrinkage used in update to prevents over-fitting." [12]
- **min split loss** - "Minimum loss reduction required to make a further partition on a leaf node of the tree." [12]
- **max depth** - "Maximum depth of a tree. Increasing this value will make the model more complex and more likely to over-fit." [12]
- **colsample by tree** - "The subsample ratio of columns when constructing each tree" [12]

The defined ranges of interest are: **learning rate** $\in \{0.05, 0.85, 0.15\}$, **min split loss** $\in \{0, 0.1, 0.2\}$, **max depth** $\in \{3, 4, 5\}$, **colsample by tree** $\in \{0.55, 0.65, 0.75\}$.

Rate	Loss	Depth	Colsample	RMSE Val	RMSE Pre
0.085	0.2	5	0.65	0.1511	0.1617
0.150	0.2	4	0.65	0.1555	0.1623
0.150	0	4	0.75	0.1561	0.1623
0.085	0.2	4	0.75	0.1536	0.1626
0.085	0.1	5	0.65	0.1519	0.1628
0.085	0.2	5	0.75	0.1523	0.1629
0.150	0.2	4	0.75	0.1523	0.1632
0.085	0.1	4	0.75	0.1536	0.1633
0.085	0.1	4	0.65	0.1527	0.1636
0.05	0.2	5	0.75	0.1527	0.1636
0.05	0.2	5	0.65	0.1523	0.1653

TABLE II
RESULTS OF HYPER-PARAMETER TESTING, XGBOOST

*Results were sorted by RMSE Pre value

Rate - learning rate

Loss - min split loss

Depth - max depth

Colsample - colsample by tree

RMSE Val - RMSE value, calculated on the validation set

RMSE Pre - RMSE value, calculated on the preliminary testing set

The training process is quite like TabNet's training process described earlier. In the first iteration XGBoost model is trained only on 10% of the training data, and then evaluated on the following 10%. In the next iteration XGBoost model is trained on the 20% of the training data set and then evaluated on the following 10%. This process takes place till the model is trained on 90% of the data set. Then the model is trained enough that we can calculate RMSE Val value, evaluating the model on the last 10% of the training data set. Afterwards XGBoost model is trained on 100% of the training data set, and then model is evaluated on the preliminary testing data set and full testing data set. It is worth mentioning that in each iteration like in the last stage of training (training on the 100% of the data set) model is basically trained from zero - created is new instance of the model, and the model gets trained. In comparison to TabNet that makes a significant difference as TabNet model uses warm training (with use of already pretrained weights as a baseline for the model training). The XGBoost model that performed best:

Type	Learning Rate	Loss	Depth	Colsample
Best	0.085	0.2	5	0.65
Default	0.3	0	6	1

TABLE III
HYPER-PARAMETERS - BEST AND DEFAULT MODELS XGBOOST

We can notice that the default model has performed significantly worse than tuned model.

F. Combination of XGBoost and TabNet models

The final model is a combination of XGBoost and TabNet models both.

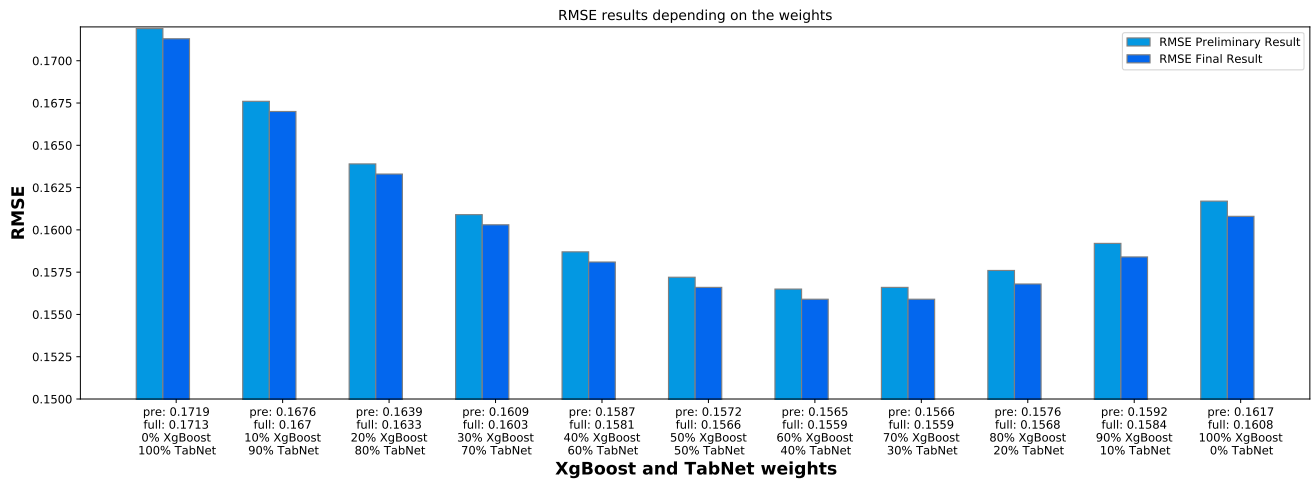


Fig. 1. Training results of XGBoost model depending on the used hyper-parameters

Type	Local Val	Preliminary Val	Final Result
Best	0.1511	0.1617	0.1608
Default	0.1739	0.2365	0.2362

TABLE IV
RESULTS - BEST AND DEFAULT MODELS XGBOOST

Results are generated independently for each one of the models. Then we use the assigned weights to generate the final results. The main problem at this stage was finding the right weights. For simplicity we have decided to check 10 different pairs. Starting with XGBoost being assigned 0 and TabNet 1, incrementing the weight assigned to the XGBoost by 0.1 and decreasing the weight assigned to TabNet by 0.1. Ultimately Fig. 1. depicts the preliminary and final evaluation of the generated results. We can notice that the best RMSE is generated for 0.6 assigned to XGBoost and 0.4 assigned to TabNet with final results of **0.1565** for preliminary validation and **0.1559** as a final result. It is worth noticing that XGBoost has significantly outperformed TabNet, with TabNet's results of **0.1719** and **0.1713** and XGBoost **0.1617** and **0.1608** as preliminary and final results respectively.

III. CONCLUSIONS

In this paper, we have introduced the approach that we have used for FedCSIS 2022 Challenge. We have described the data pre-processing, handling the missing data and two models that we have used for our solution: **XGBoost** and **TabNet**. It is also worth noticing that the process of training the XGBoost model takes much less time in comparison to the process of training the TabNet model. Although the result is worse than the baseline solution, we can notice how the combination of both models can outperform those two models working separately. Although XGBoost has significantly outperformed the TabNet model itself, the combination of both models has quite significantly outperformed both models working alone.

REFERENCES

- [1] A. Janusz, A. Jamiołkowski, and M. Okulewicz, "Predicting the costs of forwarding contracts: Analysis of data mining competition results," in *Proceedings of the 17th Conference on Computer Science and Intelligence Systems, FedCSIS 2022, Sofia, Bulgaria, September 4-7, 2022*. IEEE, 2022.
- [2] E. W. Ngai, L. Xiu, and D. C. Chau, "Application of data mining techniques in customer relationship management: A literature review and classification," *Expert systems with applications*, vol. 36, no. 2, pp. 2592–2602, 2009. [Online]. Available: <http://dx.doi.org/10.1016/j.eswa.2008.02.021>
- [3] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," *Knowledge-based systems*, vol. 46, pp. 109–132, 2013. [Online]. Available: <http://dx.doi.org/10.1016/j.knsys.2013.03.012>
- [4] I. Portugal, P. Alencar, and D. Cowan, "The use of machine learning algorithms in recommender systems: A systematic review," *Expert Systems with Applications*, vol. 97, pp. 205–227, 2018.
- [5] S. Gupta, D. Hanssens, B. Hardie, W. Kahn, V. Kumar, N. Lin, N. Ravishanker, and S. Sriram, "Modeling customer lifetime value," *Journal of service research*, vol. 9, no. 2, pp. 139–155, 2006. [Online]. Available: <http://dx.doi.org/10.1177/1094670506293810>
- [6] J. Ahn, J. Hwang, D. Kim, H. Choi, and S. Kang, "A survey on churn analysis in various business domains," *IEEE Access*, vol. 8, pp. 220 816–220 839, 2020. [Online]. Available: <http://dx.doi.org/10.1109/ACCESS.2020.3042657>
- [7] D. L. García, À. Nebot, and A. Vellido, "Intelligent data analysis approaches to churn as a business problem: a survey," *Knowledge and Information Systems*, vol. 51, no. 3, pp. 719–774, 2017. [Online]. Available: <http://dx.doi.org/10.1007/s10115-016-0995-z>
- [8] S. Ö. Arik and T. Pfister, "Tabnet: Attentive interpretable tabular learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 8, 2021, pp. 6679–6687.
- [9] L. Katzir, G. Elidan, and R. El-Yaniv, "Net-dnf: Effective deep modeling of tabular data," in *International Conference on Learning Representations*, 2020.
- [10] S. Popov, S. Morozov, and A. Babenko, "Neural oblivious decision ensembles for deep learning on tabular data," *arXiv preprint arXiv:1909.06312*, 2019.
- [11] X. Huang, A. Khetan, M. Cvitkovic, and Z. Karnin, "Tabtransformer: Tabular data modeling using contextual embeddings," *arXiv preprint arXiv:2012.06678*, 2020.
- [12] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. New York, NY, USA: ACM, 2016, pp. 785–794. [Online]. Available: <http://doi.acm.org/10.1145/2939672.2939785>

Using gradient boosting trees to predict the costs of forwarding contracts

Sławomir Pioroński

Faculty of Mathematics and Computer Science
 Adam Mickiewicz University
 Uniwersytetu Poznańskiego 4 Street
 61-614 Poznań, Poland
 Email: slawomir.pioronski@amu.edu.pl

Tomasz Górecki

Faculty of Mathematics and Computer Science
 Adam Mickiewicz University
 Uniwersytetu Poznańskiego 4 Street
 61-614 Poznań, Poland
 Email: tomasz.gorecki@amu.edu.pl

Abstract—When selling goods abroad or bringing them into the country from foreign partners, we face the problem of delivery. The division of responsibilities related to this between the manufacturer and the recipient sometimes varies. In such a situation, it is reasonable to use the services of a forwarding company. Then a forwarding contract is concluded, which specifies the details of the service, but the most important issue remains the selection of its price. In this paper, we present results obtained using the LightGBM method on the forwarding contracts pricing challenge held as part of the FedCSIS 2022 conference.

Index Terms—data mining competition, LightGBM, forwarding contracts

I. INTRODUCTION

A FORWARDING contract is a contract in which one of its parties – the forwarder undertakes to perform various services related to transportation in the course of his own business, such as sending or receiving a shipment, and the other party – to pay remuneration in return. A good forwarder will not only efficiently organize transportation but will also help reduce the cost of the transaction. However, to remain price competitive and still make a profit, he must have a good tool for predicting the cost of executing such a contract.

To predict forwarding contract costs based on tabular data we can use various machine learning methods [1]. We decided to use a gradient boosting algorithm, especially LightGBM, because of its speed and accuracy.

The rest of this paper is structured as follows. We first present the related work and then we give a short description of FedCSIS 2022 challenge. In Section IV we describe the data processing steps. Section V contains the description of the model used in our experiment. Next section presents results. Finally, in Section VII we summarize the findings and discuss possible future work.

II. RELATED WORK

Gradient boosting is a machine learning technique, which produces a prediction model in the form of an ensemble of weak prediction models, mainly decision trees. It creates the model like other boosting methods do, but it generalizes them by allowing optimization of an arbitrary differentiable loss function. Gradient boosting was first presented in 1997 [2],

and has been refined over the last decade. There are many different implementations:

- XGBoost – an algorithm written by Tianqi Chen [3]. Probably the best known and most used implementation.
- LightGBM – Microsoft’s algorithm [4].
- Catboost – an algorithm by the Russian company Yandex, designed to deal with categorical data [5].

LightGBM has many of XGBoost’s advantages, including sparse optimization, parallel training, multiple loss functions, regularization, bagging, and early stopping. A main difference between the two lies in the construction of trees. LightGBM does not grow a tree level-wise – row by row. Instead it grows trees leaf-wise. Besides, LightGBM does not use the sorted-based decision tree learning algorithm as XGBoost. Instead, it implements a highly optimized histogram-based decision tree learning algorithm, which yields great advantages on both efficiency and memory consumption. The LightGBM algorithm utilizes two novel techniques called Gradient-Based One-Side Sampling and Exclusive Feature Bundling which allow the algorithm to run significantly faster while maintaining a high level of accuracy.

We can find many examples of LightGBM being used in machine learning competitions [6].

III. FEDCSIS 2022 CHALLENGE

A. Data

The available training data set [7] contains a five-year history of contracts accepted by a Polish company. It consists of two main tables. The first one contains basic information about the contracts (36 features), and the second one describes the main sections of the planned routes associated with each contract (60 features). In the train set, we have 330 055 contracts and in the test set, we have 72 452 contracts. In addition, participants had an additional table (4 features) containing historical wholesale fuel prices.

B. Task

The theme of the competition was to forecast the costs associated with the execution of forwarding contracts, based on data from contracts and planned routes. The goal of the

competition was to prepare and develop a model that forecasts the costs of individual orders as accurately as possible.

C. Evaluation

The solutions were assessed by the root mean square error:

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2},$$

where y_i is an observed value, \hat{y}_i is a predicted value and n is the number of data records in the data set.

Initial scores were evaluated via the KnowledgePit online platform [8] and published on a challenge leaderboard calculated on a small subset of the test set fixed for all participants. The final score was published after the challenge using the remainder of the test data set.

IV. DATA PROCESSING

Data from both main tables were used for training. Since the data in the second table contains at least 2 rows for each contract (one row for each route step), it needs to be processed accordingly. In this case, the choice was made to include information for step two of the planned route with each contract. In addition, average monthly fuel prices are added for each contract, calculated from an additional third table, based on the month the route began. This resulted in a training data size of $330\ 055 \times 98$. Descriptions of each column can be found on the competition page [8].

A. Adding new features

Intuitively, an important factor affecting the price of the contract is the route duration, so a new column was created: `hours_diff` – the difference between `route_end_datetime` and `route_start_datetime` expressed in hours. The natural logarithms (with a lower bound equal to -1) of the following features were then created: `hours_diff`, `km_total`, `km_nonempty`, `km_empty`, `train_km`. Pearson correlation coefficients increased significantly for the first three characteristics listed. For example, for `hours_diff` it is 0.63, but for its logarithmic version, we get 0.89.

The numerical columns `train_intervals` and `ferry_intervals` were transformed into categorical variables with values of "0", "1", and "2+", which correspond to their numerical values.

At this point, it is worth mentioning that we used the Microsoft's FLAML library [9] (version 1.0.1). It contains many facilities, one of which is the automatic generation of the following numeric features for each datetime feature: year, month, day, minute, second, day of the week, day of the year, and quarter.

B. Repairing route datetime data

Real-world data from companies typically contains human errors. This case is no different. By checking in how many cases `route_start_datetime` is later than `route_end_datetime` we get 47 (0.01%) samples in the training set and as many as 405 (0.56%) in the test set. We see that for the training set, this is a

very small number of examples that could easily be removed. However, in the case of the test set, this could determine the outcome of the competition.

Unfortunately, we don't know what this data should look like. Nevertheless, in reviewing this data, there are two types of errors:

- 1) Dates are in the wrong order.
- 2) The month or day was entered incorrectly.

The `route_start_datetime` and `route_end_datetime` are based on the `estimated_time` column from the second table, which contains information about the planned time of arrival at a given route step point, so it was the values from this column that were corrected. This was done with a simple script and then verified manually. Incorrect datetimes typically occurred for the first 2 or 3 steps within a given contract and were the same with date accuracy (no time information). For each contract, initial datetimes equal in date were selected. For simplicity, we will use a single date. In the first case the difference between the selected date with the last date was checked, if they were less than 14 days then selected datetimes were moved to the last places. If a type 1 error was not detected, the presence of a type 2 error was checked. Here we compared the number of days for the selected date and the next date after it (let's call it a comparison date). Assuming we have 1 after 31 and each month has 31 days, then if all we had to do was increase the number of days of the selected date by a maximum of 7 to get the number of days of the comparison date, we set the month and year of the selected datetimes to the largest possible so as not to exceed the comparison date. Any other instances, e.g., incorrectly entered days, were corrected manually through individual decisions.

C. Deleting some data

Often, some of the data we have is unnecessary and may even degrade the performance of the model. By comparing the values of the `id_currency` column, we can see that the training set has 6 unique additional values than the test set. Similarly, for the `step_type` feature from the second table, we obtain one unique redundant value in the training set. The temperature column, which reports the temperature level in a refrigerated trailer, was removed due to the high percentage of missing values (89% in both the training and test sets) and the way the values in it were recorded. This is string-type data with individual temperatures, temperature ranges, or additional information about the cooling mode. We have 760 unique values in the training set, 194 in the test set, and 99 values common to both sets.

After removing such contracts from the training set, and after removing the `id_contract` and temperature columns, we obtain a data set of size $329\ 349 \times 102$. These operations did not result in a large reduction of the data set, as the number of rows decreased by only 0.21%, but we still got rid of unnecessary data.

FLAML automatically discards features with constant values during training. In this case, we had 5 such features, all from the second table: `step`, `train`, `train_km`, `train_line`, `tail_fin`.

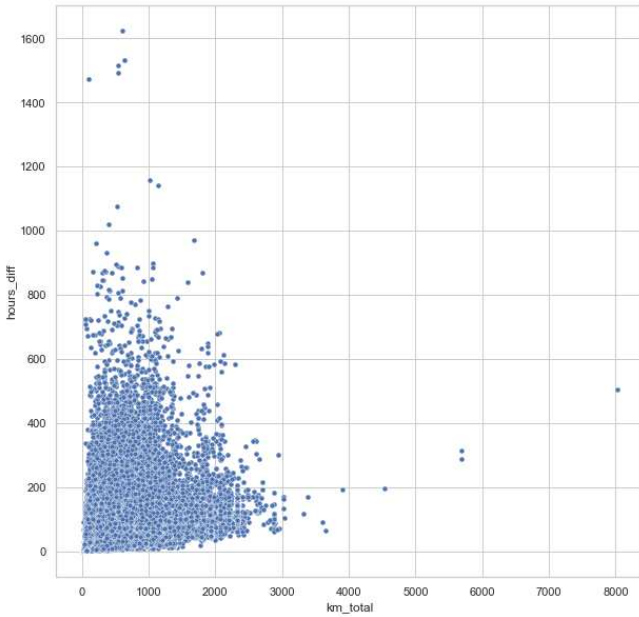


Fig. 1. Relationship between km_total and hours_diff (test set).

V. MODEL

A. Training strategy

LightGBM models, among other things, automatically perform feature selection and handle missing values. So what remains is the selection of appropriate hyperparameters. The FLAML library has solutions for this purpose. It offers two methods to optimize hyperparameters: CFO [10] and Blend-Search [11]. Both methods require a low-cost initial point from which the search begins if such a point exists. In the case of the former, the search gradually moves toward higher-cost regions if needed. It is a local search. The latter method combines the local with global search, i.e., it starts checking new starting points before the local search reaches full convergence. We used the “auto” mode to select the method, which should result in the selection of the CFO.

B. Output scaling

Since the training set contained data from 2016-01 to 2020-10 and the test set from 2020-09 to 2021-11, we decided to scale the model output. Outputs for the 2020 test data were multiplied by 0.998, while outputs for subsequent months of 2021 were multiplied by 1.001, 1.002, up to 1.011.

A measure like RMSE is sensitive to single, large errors. Thus, model errors for outliers can significantly degrade the final result, and the occurrence of outliers can be easily observed in Fig. 1. Hence, the idea was born to additionally scale values for samples that may be outliers. Thus, in the next step, for the selected points suspected of being outliers, the corresponding model results were multiplied by 1.01.

C. Outlier detection

Some of the most popular methods for finding outliers are isolation forest [12] and local outlier factor [13]. An isolation forest was chosen because of its shorter operating time.

In this case, it is necessary to pass only numeric type features to the model. The datetime type features have been removed and the corresponding year, quarter, month, and day of week features have been added in their place. In addition, the data set was expanded to include values for step one and last step from the second table (in some cases step two = last step). Next, columns with constant values and categorical features with more than 5% of missing values were removed. In other cases, they were imputed with the most frequent value. 2 of them were two-valued, so they were mapped to binary values. The Target Encoder was then fitted on the training set, i.e. the average target value for each class was taken.

The popular open-source machine learning library scikit-learn [14] (version 1.0.2) was used to train the model (with default hyperparameters). Mentioning this is important for the interpretation of the results since in this implementation the scores returned by the isolation forest can take negative values.

For points for which the isolation forest score was less than -0.05 , were from 2021, and hours_diff was greater than 50 the final values were increased an additional 1%.

VI. EXPERIMENTAL RESULTS

A. Hyperparameter tuning

The final solution was created using an Intel Core i5-8300H processor and 16 GB of RAM (DDR4, 2400 MHz) on Microsoft Windows 10. It was found after 1617 seconds, with the search time set at 1800 seconds. During this time, the model scored 12 times better on the validation set (10% of the training set). To evaluate the quality of the model, we used the same measure as in the competition, the RMSE. After 33 seconds, we reached an RMSE of 0.1438, and the final model obtained an RMSE of 0.1298. Increasing the search time to 2700-3600 seconds can achieve an RMSE of 0.1265, but such models gave very poor results on the public competition test set, for instance: RMSE = 0.1530.

B. Obtained model

The selected model consists of 5243 trees, with a maximum number of 509 leaves in a single tree, the exact values of all tuned hyperparameters [9] can be found in Table I. Unfortunately, due to the size of the trees, we cannot visualize them in a meaningful way, but for practical reasons, the importance of individual features during prediction is useful. For this purpose, we will use total gain [3]. The features with the highest values are shown in Fig. 2. Unsurprisingly, we see that the distance traveled and the route time have a very large impact on the final prediction result.

On the public part of the test set, the model achieved a score of 0.1458. Here we can see that correcting the dates in Section IV-B was the right thing to do, because without it we achieved a score of 0.1469.

TABLE I
FINAL HYPERPARAMETERS (TO FIVE DECIMAL PLACES).

Hyperparameter	Value
n_estimators	5243
num_leaves	509
min_child_samples	8
learning_rate	0.01079
log_max_bin	6
colsample_bytree	0.34362
reg_alpha	0.00198
reg_lambda	0.01343

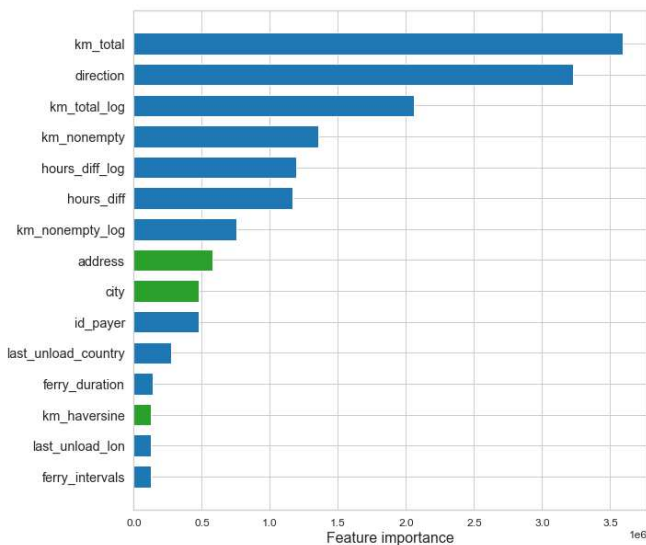


Fig. 2. The 15 features with the highest value of total gain. The features in the first table have been colored in blue, and those in the second in green.

C. Model output handling

The scaling idea was born out of the large discrepancy between the results on random subsets of the training data and the public portion of the test set. After using scaling of the model output described in Section V-B to the model from Section VI-B, our result improved. In this configuration, the RMSE on the public part of the test set is 0.1438. As can be seen, the intuition was correct, since linear dependencies with a precision of one month should be easily learned by the model, so they were not present in the training data. The reason may have been inflation was not openly recorded in the data.

Since such simple scaling does not exhaust the possibilities for improving the score, we used the isolation forest to look for values worth further improving. As a result, the RMSE changed from 0.1438 to 0.1434.

VII. CONCLUSIONS

This paper presents a powerful regression model that can deliver excellent predictions of the costs of forwarding contracts. We chose the LightGBM, because of its simplicity and

speed. The model achieved the performance of the RMSE score of 0.1420 on a test set placing fourth (out of more than 50 teams that added a total of nearly 2 000 correctly formatted solutions) in the FedCSiS'2022 competition. Our model lost by only 2.606% to the best solution and lost only 0.9155% to third place. At the same time, it was better than the baseline model (baseline model placed fifth) by 3.873%. Worth adding, that the model got better final results than the results on the preliminary data set (0.1434). In addition, it is worth emphasizing that all calculations were performed in less than an hour on the average CPU.

The solution is pretty simple, so we have a lot of options here in terms of future work. For example, the topic of scaling model output is not exhausted. The idea of scaling outputs for points suspected of being outliers came up on the last day of the competition, so this was tested very briefly. So it is worth checking the outputs for other points suspected of being outliers and other multipliers. Taking it a step further, it is possible to see if the multiplier can be calculated for each point separately depending on some features.

REFERENCES

- [1] T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd Edition*, ser. Springer Series in Statistics. Springer, 2009. ISBN 9780387848570
- [2] L. Breiman, "Arcing the edge," 1997.
- [3] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. New York, NY, USA: ACM, 2016. doi: 10.1145/2939672.2939785 pp. 785–794.
- [4] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., 2017.
- [5] A. V. Dorogush, V. Ershov, and A. Gulin, "Catboost: gradient boosting with categorical features support," *ArXiv*, vol. abs/1810.11363, 2018. doi: 10.48550/ARXIV.1810.11363
- [6] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, "M5 accuracy competition: Results, findings, and conclusions," *International Journal of Forecasting*, 2022. doi: 10.1016/j.ijforecast.2021.11.013
- [7] A. Janusz, A. Jamiołkowski, and M. Okulewicz, "Predicting the costs of forwarding contracts: Analysis of data mining competition results," in *Proceedings of the 17th Conference on Computer Science and Intelligence Systems, FedCSIS 2022, Sofia, Bulgaria, September 4-7, 2022*. IEEE, 2022.
- [8] "Fedcsis 2022 challenge: Predicting the costs of forwarding contracts," <https://knowledgepit.ml/fedcsis-2022-challenge/>, accessed: 2022-06-20.
- [9] C. Wang, Q. Wu, M. Weimer, and E. Zhu, "Flaml: A fast and lightweight autml library," in *MLSys*, 2021.
- [10] Q. Wu, C. Wang, and S. Huang, "Frugal optimization for cost-related hyperparameters," in *AAAI'21*, 2021.
- [11] C. Wang, Q. Wu, S. Huang, and A. Saied, "Economical hyperparameter optimization with blended search strategy," in *ICLR'21*, 2021.
- [12] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation forest," in *2008 Eighth IEEE International Conference on Data Mining*, 2008. doi: 10.1109/ICDM.2008.17 pp. 413–422.
- [13] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "Lof: Identifying density-based local outliers," in *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, ser. SIGMOD '00. New York, NY, USA: Association for Computing Machinery, 2000. doi: 10.1145/342009.335388 pp. 93–104.
- [14] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

Predicting the Costs of Forwarding Contracts Using XGBoost and a Deep Neural Network

Łukasz Podlowski, Marek Kozłowski
 National Information Processing Institute
 Laboratory of Natural Language Processing
 al. Niepodległości 188b 00-608 Warsaw, Poland
 Email: lpodlowski@opi.org.pl; mkozłowski@opi.org.pl

Abstract—This article presents an application of an XGBoost and deep neural network ensemble as a solution for a task assigned at the FedCSIS 2022 Challenge: Predicting the Costs of Forwarding Contracts. We demonstrate that prediction quality can be improved by combining the two approaches. We present a neural network architecture based on three independent flows. We then discuss the influence of long short-term memory units on the risk of overfitting. Finally, we show that the static XGBoost model can complement a neural network that processes dynamic data.

Index Terms—XGBoost; deep neural network; LSTM; Costs of Forwarding Contracts

I. INTRODUCTION

COST estimations are an imperative factor in business decisions in the transport sector. Researchers have demonstrated that due to a variety of dependencies, the price estimation of shipments in the shipping industry is frequently complex. Based on these studies, shipment pricing methods can be classified into two major classes: scenario-based pricing methods and algorithmic pricing. This article focuses on the former.

We concentrate on the logistical problem of predicting the costs of executing forwarding contracts. Our work relates closely to the FedCSIS 2022 Challenge: Predicting the Costs of Forwarding Contracts¹, which was organised in association with the Conference on Computer Science and Information Systems (<https://fedcsis.org/>). The competitors were tasked with designing high-quality methods for predicting the costs associated with forwarding contracts based on many features that describe key contract data and planned routes.

The competitors' task involved developing a predictive model that assessed the actual costs of individual orders as accurately as possible. Such models will be used in the future to support freight forwarders in the selection of profitable contracts.

II. RELATED WORK

The need for transportation arises from the need to move goods from one place to another in line with consumer demand. Sea transportation is one of the cheapest and oldest modes of transportation of goods. During the last twenty years, seaborne trade has accounted for approximately 80% of the

world's and 90% of developing countries' total trade volume. According to recent statistics, since the 1980s, global seaborne trade has increased in size almost threefold, and containerised trade has increased also [1]. The goods delivered to major sea ports are next distributed by road or by rail. The global weight of loads in ports and the density of road/railway link networks continues to increase steadily. This means that the problem of optimal planned deliveries has become a significant challenge.

The article of Joo, Min, and Smith [2] presents an entire framework for benchmarking freight rates based on current and historical data. This is one of the first articles to examine shipping cost differentials between different shippers and to determine their causes.

The authors of [3] address the research gap of optimal spot shipment price calculation based on current shipment demand and available shipping capacity. The authors gathered data from various sources to generate a shipping dataset for 2016–2018. They use regression and correlation analysis to quantify their research outcomes.

A fuzzy regression forecasting model was introduced in [4] to forecast demand by examining the current international air cargo market. The difficulty of such forecasting derives primarily from individuals' differing perceptions of their socioeconomic environments and their competitiveness when evaluating risk. The main purpose of using fuzzy regression is to resolve these uncertainties and specificities while accommodating individuality.

Many long-term and long-distance transportation services are offered now via various types of auction; firms in the sector must deliver competitive prices if they want to win the competitions. The article of Nataraj et al [5] explores the application of forecasting and statistical learning methods to enhance the competitiveness of firms when applying for tenders. The authors use time series analysis to: (i) forecast the long-term cost of logistics services; and (ii) construct 'risk-aware' intervals for the prices to be offered in bids.

Yang and Mehmed [6] adopt an artificial neural network to forecast shipping freight rates. The key objective of their work is to improve the forecasting accuracy of traditional time series analysis. They evaluate the accuracy of their forecasting models using the mean square error with historical data. The authors used two different dynamic artificial neural network models, NARNET and NARXNET, and compared their per-

¹<https://knowledgepit.ml/fedcsis-2022-challenge/>

formance. The experimental results suggest that, in general, NARXNET outperforms NARNET in all forecast horizons. This reveals the importance of the information contained in forward freight agreements in improving forecasting accuracy.

In 2022, Adrian Viellechner defended his PhD dissertation [7] on the application of machine learning models to transportation forecasting. First, Viellechner analyses the container shipping industry to predict delays of vessels. In his second study, he presents how machine learning methods can be applied to predict spot rates in container shipping. With accuracy of 89%, his forecasts support the decision-making of various shipping players in the negotiation of transportation contracts. By proposing prediction solutions that feature high accuracy, robustness, and applicability in practice, Viellechner's dissertation demonstrates that machine-learning-based solutions enhance effectiveness in transportation.

III. PROBLEM DESCRIPTION

The problem involves predicting the current costs of individual orders using detailed information, such as the type of order, the basic characteristics of the shipped goods (e.g. dimensions, special requirements), and the expected route that a driver would traverse. Using machine learning approaches, our goal is to create an accurate regressive method for predicting the costs associated with forwarding contracts; one that is based on contract data and planned routes. We evaluate the regression models using the root mean square error measure.

IV. DATA

The accuracy of the forwarding contract cost prediction method depends heavily on the quality and quantity of the training data.

The datasets made available to the competitors contained six years of order history that appeared on the transport exchange, along with details such as the type of order, the basic characteristics of the shipped goods (e.g. dimensions, special requirements), and the expected route that a driver would traverse. More details about competition and data are presented in [8].

V. DATA PROCESSING

A. Data preprocessing

The initial step of the data's preprocessing involved translating categorical features into one-hot encoding. We set the minimum threshold for any categorical value at 2,500; if the value occurred fewer times than this threshold, we translated it as 'unknown'. We used a regex-based method to extract minimal and maximal temperatures. When information about temperature was unavailable, we filled it in with a constant value of 35. We assumed that high temperatures did not increase contract costs, and that 35 was a neutral value. Since the dataset does not contain this value in the temperature column, it allows the model to handle situations when the absence of temperature data is relevant for undisclosed reasons. Our future experiments showed temperature information increased prediction quality. Based on date information associated with

journey start dates, we matched fuel prices and added them as an additional column to each row.

We will refer to the dataset outlined above as our base set. We adopted a different approach for the neural network and XGBoost models to use the additional data included in the sequences of steps corresponding to each row from the base set. For XGBoost, we aggregated additional information, such as how many times an external fleet was used in the transport process, or how many times a specific country was visited. For the neural network, we removed some columns from the sequence data, such as cargo hold height and width, and treated this data as a fixed-size window of twelve steps. Approximately 0.4% of the sequences included in training set were longer, which forced us to cut off the sequences' tails to a fixed twelve-length size. Approximately 99.6% of sequences were shorter; to solve this difficulty, we padded the sequences with zero vectors. We choose sequences size respect to balance between computation complexity and prediction quality. We found out based on the cross-validation that lower sequence size increased prediction error.

Our first experiments revealed large differences between the cross-validation results and the preliminary scores. This led us to the conclusion that our models were overfitted. These differences disappeared, however, after we disabled the shuffling of the data during the cross-validation procedure. We supposed that the dates included in the training set could lead to the model becoming overfitted.

We assumed that date relations with target values would suppress the recognition of other patterns. To avoid this situation in our subsequent experiments, we prepared an additional set with the date information removed to make it more difficult for the model to build a relation between start times and predicted expenses.

VI. PREDICTION MODELS

We used two different models: XGBoost and a deep neural network(DNN). We intended to compare the approaches and to verify whether the two models could be complementary in the regression problem embedded in sparse space. Significant differences exist in the training data prepared for each model. XGBoost naturally models static data and cannot handle dynamic data directly. For this reason, all information from the sequences is aggregated then presented as scalar values for each row from the base set. This method introduces the risk of eliminating important information from the data included in the order of steps in a sequence.

The deep neural network can handle dynamic data using a recurrent approach. We used popular long short-term memory units, which are known for their high efficiency and have proved their worth in a wide range of fields, including machine translation [9], financial market forecasting [10] and air quality prediction [11].

Long short-term memory is commonly used with one-hot encoding in various problems, such as electric load forecasting [12] and the construction of intelligent agents that play video games [13]. We used long short-term memory to enable the

models to extract important information from the order of sequences, encode it in a static space, and use it with the static data included in the base set.

We encoded the date as a number of minutes starting from 1st January 2016. We allowed XGBoost to operate on date information; we removed this information from the dataset of the deep neural network, however. For the final prediction, we used the strategy of split data set into 75% train and 25% test. For XGBoost this split was done randomly but for a DNN we used the oldest 25% of data as a validation set.

A. XGBoost

From a theoretical perspective, tree-based methods should naturally fit to data with mixed domain values. This suggests that any ensemble of trees is an effective choice for concatenate categorical features with other domains, such as the distance between two cities or temperature. Tree-based methods are limited to static data, which could lead to the loss of important information from the sequences. We accepted that risk and selected XGBoost as our base model.

Our first experiments were based on a fourfold cross-validation method. We achieved an impressive root mean square deviation of 0.1324. However, our preliminary result, evaluated on a small part of the test set, was 0.4047, which was significantly worse. In this paper we call this model XGBoost CV-optimized. We observed that the root mean square deviation based on cross-validation increased to 0.158 when we disabled data shuffling. This suggests that the data contained some undisclosed information associated with time, which led to overfitting. From this point, we based each step only on 75% of the data; we used the remainder as a validation set for the training procedure. The experimental values are presented in Table I.

To prevent overfitting, we explored the possibility of increasing the model's regularisation parameters. We decreased subsampling to 0.8, which forced the model to randomly skip some rows in training iterations. We also increased the regularisation λ parameter, which corresponds to L2 regularisation. This approach forced our model to become more conservative and less likely to overfit. Adjusting both parameters decreased the gap between the final score and the cross-validation result. Our experiments failed to reveal any opportunity to increase the α parameter, which corresponds to L1 regularisation without significantly lowering the quality of the model's predictions.

To find a reason for the high error of XGBoost CV-optimized predictions, we compared feature importance tables of our models. We recognized that the CV-optimized model increased the importance of specific categorical values. The biggest difference was the importance of the 'unknown' *prim ferry line* value generated by the procedure described in V. Importance for this value increased from 0.0001 to 0.08. We also noticed that XGBoost CV-optimized tended to be more confident on every localization feature corresponding to country code like *route start country* or *first unload country*.

B. Deep neural network

Our second model was based on a neural network. We decided to allow fully-connected layers to encode the inputs before we concatenated them into fully-connected layers operating on the whole data. The input was divided into three independent flows:

1) *localisation data*: static data from the base set, including all information about geolocalisation associated with route starts and ends, as well as first loadings and last unloadings. These vectors contained latitudes, longitudes, and country codes.

2) *general data*: static data from the base set, including all information not included in the localisation data.

3) *sequence data*: vectors that represent fixed-size sequences of route steps. Applying long short-term memory units enabled us to handle dynamic data and to transform sequences to static fully-connected rectifier layers.

To prevent overfitting and the local minima trap, we added small Gaussian noise layers to the inputs. These layers guarantee the training process to be out of balance, which prevents the local minima trap. The amount of information in a weight can be controlled by adding Gaussian noise and the noise level can be adapted during learning to optimise the trade-off between the expected squared error and the information in the weights' [14]. Adding noise to the input also enabled us to control the degree of fitting to the data. Additionally, we used dropout with $p = 0.4$ to each layer, which decreased the probability of the model's overfitting [15]. The network architecture is presented in Figure 1.

For the training network, we used the Adam optimiser [16]. Adam automatically adapts parameters to training, which enabled us to avoid the arduous process of training tuning on the mixed domains. We repeated the training process after the competition ended to gain greater insight into the deep neural network's results. For this training, we used all of the testing data as a validation set. We present a chart of root mean square errors during the training process in Figure 2. We can observe that the training process is highly chaotic initially, and that the validation root mean square errors rapidly reduce to 0.158 – 0.16 before stabilising. It is noteworthy that the chart begins from the twentieth epoch. The best result achieved in this experiment was 0.1564.

To analyse the influence of the sequential data flow processed using long short-term memory units, we prepared an additional deep neural network with similar architecture, but without the sequential data flow. The error observed in the training process is presented in Figure 3. We made two key observations associated with the overfitting risk:

- The deep neural network with long short-term memory tends, as it continues to train, to enlarge the gap between training and the validation data much faster than the network without sequential data flow. This could be the result of the network's increased capacity, or could suggest that information about the order of the sequences misdirects the training process. Both reasons lead easily to overfitting.

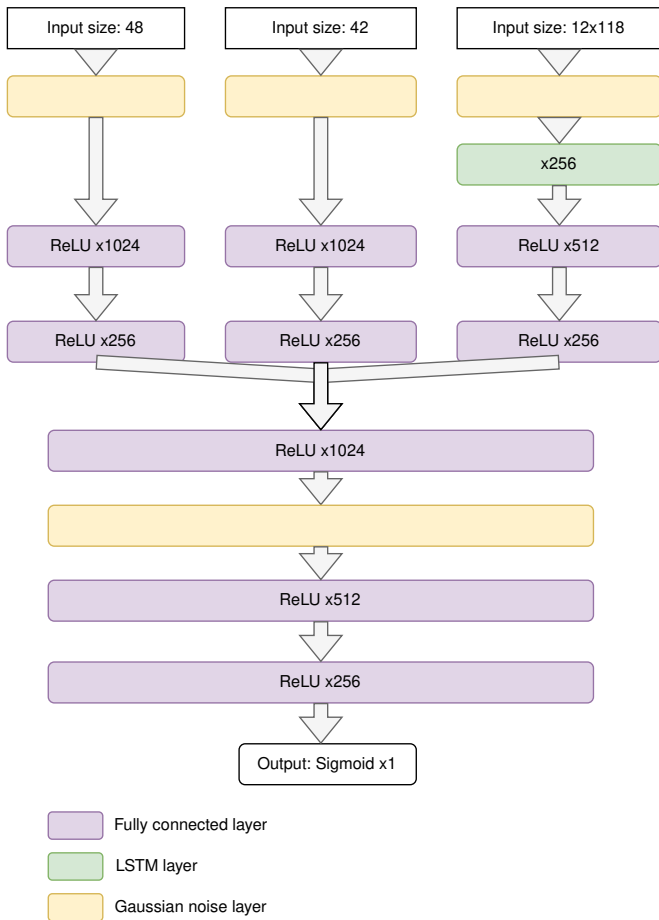


Fig. 1. The architecture of the deep neural network used as one of the solutions to a defined problem.

- The deep neural network without long short-term memory tends to gain and lose the optimisation target in each epoch in an increasingly synchronised manner. The shapes of the training and validation error curves correlate more closely on this network.

We conclude that long short-term memory would extract important information from the dataset and increase prediction quality but should be used carefully because of the overfitting problem.

VII. RESULTS

To generate the final prediction, we used an ensemble of the XGBoost and deep neural network models as the arithmetic mean of both predictions. As illustrated in Table I and Figure 4, this ensemble reaches the best result of 0.1498. This root mean square error value is significantly better than the results of the individual models, demonstrating clearly that the deep neural network and XGBoost extracted different information from the complementary data. In the training process, we used the splitting data strategy described in VI. This strategy lets us have better control over the overfitting.

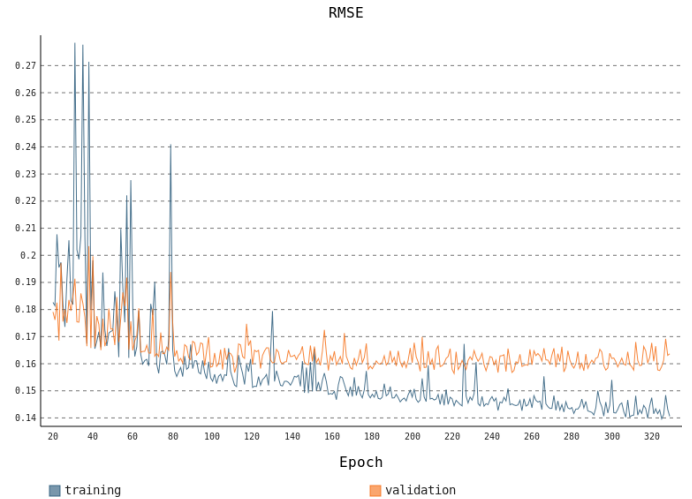


Fig. 2. The root mean square errors during the deep neural network's training process. The blue series represents error on the training data; the orange series represents error on the whole validation set—which corresponds directly to the final scores of the competition.

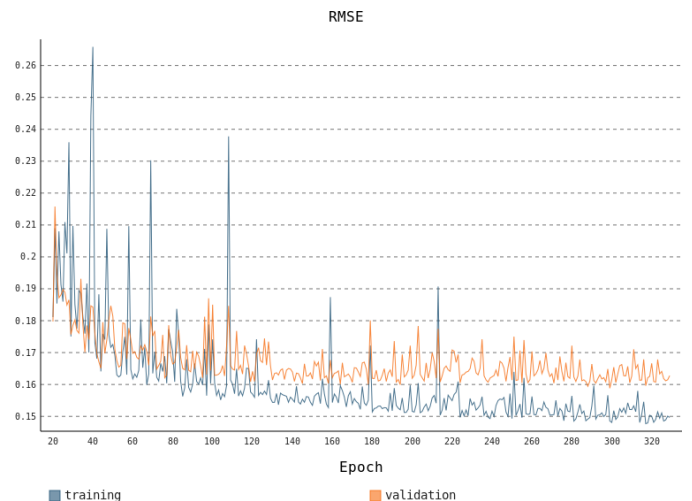


Fig. 3. The root mean square errors during the training process of the neural network without sequential data flow. The blue series represents error on the training data; the orange series represents error on the whole validation set—which corresponds directly to the final scores of the competition.

Our cross-validation-optimised XGBoost model reaches a root mean square error of only 0.404, which is significantly worse than the other models. We failed to identify precisely which factor caused the overfitting. Future experiments that focus on the analysis of smaller parts of the dataset should be performed to gather detailed information on this problem.

VIII. CONCLUSIONS

This article presents our solution to the FedCSIS 2022 Challenge: Predicting the Costs of Forwarding Contracts. During our experiments, we identified an overfitting problem, which had a significant impact on subsequent steps. We prepared two approaches: XGBoost and a deep neural network. We demonstrated that the application of a long short-term memory

TABLE I

EXPERIMENTAL RESULTS: PREDICTION QUALITY ROOT MEAN SQUARE ERRORS BASED ON CROSS-VALIDATION AND PRELIMINARY SCORES

Model	CV	Preliminary	Final score
XGBoost	0.153	0.1522	0.15252
XGBoost CV-optimized	0.1324	0.4047	0.40394
Deep Neural Network	0.1581	0.1585	0.15815
Ensemble	-	0.1500	0.14978

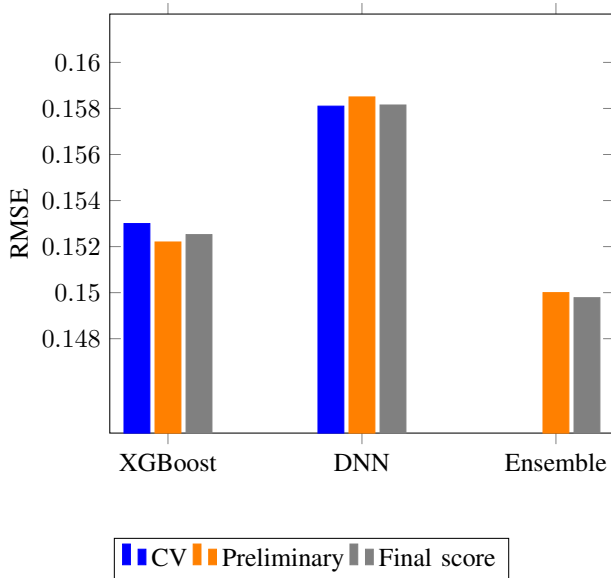


Fig. 4. Root mean square errors based on three experiments: fourfold cross-validation (blue), preliminary scores, and final scores.

layer to sequential data improves the results; however, it also increases the probability of overfitting. We demonstrated that combination of the deep neural network and XGBoost is potentially complementary, significantly increases prediction quality, and serves as a possible solution for mixing dynamic data with the static XGBoost approach.

REFERENCES

[1] Z. H. Munim and H.-J. Schramm, "Forecasting container shipping freight rates for the far east–northern europe trade lane," *Maritime*

Economics & Logistics, vol. 19, no. 1, pp. 106–125, 2017.

[2] S.-J. Joo, H. Min, and C. Smith, "Benchmarking freight rates and procuring cost-attractive transportation services," *The International Journal of Logistics Management*, 2017.

[3] A. Ubaid, F. Hussain, and J. Charles, "Modeling shipment spot pricing in the australian container shipping industry: case of asia-oceania trade lane," *Knowledge-based systems*, vol. 210, p. 106483, 2020.

[4] T.-Y. Chou, G.-S. Liang, and T.-C. Han, "Application of fuzzy regression on air cargo volume forecast," *Quality & Quantity*, vol. 45, no. 6, pp. 1539–1550, 2011.

[5] S. Nataraj, C. Alvarez, L. Sada, A. Juan, J. Panadero, and C. Bayliss, "Applying statistical learning methods for forecasting prices and enhancing the probability of success in logistics tenders," *Transportation Research Procedia*, vol. 47, pp. 529–536, 2020.

[6] Z. Yang and E. E. Mehmed, "Artificial neural networks in freight rate forecasting," *Maritime Economics & Logistics*, vol. 21, no. 3, pp. 390–414, 2019.

[7] A. M. Viellechner, "The new era of predictive analytics in container shipping and air cargo," Ph.D. dissertation, WHU-Otto Beisheim School of Management, 2022.

[8] A. Janusz, A. Jamiolkowski, and M. Okulewicz, "Predicting the costs of forwarding contracts: Analysis of data mining competition results," in *Proceedings of the 17th Conference on Computer Science and Intelligence Systems, FedCSIS 2022, Sofia, Bulgaria, September 4-7, 2022*. IEEE, 2022.

[9] G. Tiwari, A. Sharma, A. Sahotra, and R. Kapoor, "English-hindi neural machine translation-lstm seq2seq and convs2s," in *2020 International Conference on Communication and Signal Processing (ICCSP)*, 2020. doi: 10.1109/ICCSP48568.2020.9182117 pp. 871–875.

[10] A. H. Bukhari, M. A. Z. Raja, M. Sulaiman, S. Islam, M. Shoaib, and P. Kumam, "Fractional neuro-sequential arfima-lstm for financial market forecasting," *IEEE Access*, vol. 8, pp. 71 326–71 338, 2020. doi: 10.1109/ACCESS.2020.2985763

[11] J. Wang, J. Li, X. Wang, J. Wang, and M. Huang, "Air quality prediction using ct-lstm," *Neural Computing and Applications*, vol. 33, pp. 1–14, 05 2021. doi: 10.1007/s00521-020-05535-w

[12] A. Janusz, T. Tajmayer, and M. Świechowski, "Helping ai to play hearthstone: Aaia'17 data mining challenge," in *2017 Federated Conference on Computer Science and Information Systems (FedCSIS)*, 2017. doi: 10.15439/2017F573 pp. 121–125.

[13] K. H. Kim, B. Chang, and H. K. Choi, "Deep learning based short-term electric load forecasting models using one-hot encoding," *Journal of IKEEE*, vol. 23, no. 3, pp. 852–857, 2019.

[14] G. E. Hinton and D. v. Camp, "Keeping neural networks simple," in *International Conference on Artificial Neural Networks*. Springer, 1993, pp. 11–18.

[15] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, Jan. 2014. [Online]. Available: <http://dl.acm.org/citation.cfm?id=2627435.2670313>

[16] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: <http://arxiv.org/abs/1412.6980>

Diversified gradient boosting ensembles for prediction of the cost of forwarding contracts

Dymitr Ruta, Ming Liu, Ling Cen
 EBTIC, Khalifa University, UAE
 {dymitr.ruta,liu.ming,cen.ling}@ku.ac.ae

Quang Hieu Vu
 ZaloPay, VNG Corporation, Vietnam
 hieuvq@vng.com.vn

Abstract—A common business practice for transportation forwarders is to bid for shipping contracts at the transport or freight exchanges. Based on the detailed contract requirements they try to estimate the total expected cost of its execution and accordingly bid with the fixed price in advance for delivering such shipping service at the prescribed specification and schedule. The capability to accurately predict the cost of contract execution is the critical factor deciding about the profitability of offered shipping services as well as the amount of business drawn from freight exchanges. However, given highly volatile nature of the transport services ecosystem, it is difficult to simultaneously account for countless dynamically changing factors like fuel prices, currency exchange rates, temporal and spatial multitude of routing and implied traffic risks, the properties of cargo and shipping vehicles etc., which leads to big cost under- or over-estimation resulting with loss-making contracts or equally painful missed revenue opportunities. In the context of FedCSIS 2022 data mining competition we propose an accurate and robust predictor of the cost of forwarding contracts built upon the detailed contract data using the ensemble of the state-of-the-art gradient boosting-based regression models. Our established feature engineering framework combined with deep parametric optimization of the individual models and multi-faceted diversification techniques guiding hybrid final model ensembles were instrumental to outperform all the competitive predictors and win the FedCSIS 2022 contest.

Index Terms—Cost Prediction of Forwarding Contracts, Gradient Boosting Trees, CatBoost, XGBoost, LightGBM, Stacking, Diversity, Model Diversification, Ensemble Learning.

I. INTRODUCTION

WITH the development of IoT (Internet of things), e-commerce and continuous globalization, the business providing logistics service and involving supply chain for supply chain planning functions or transport management has become increasingly important. Big data analytics for intelligent transportation and prediction analysis with machine learning techniques have boosted management of transportation and logistics by providing intelligent solutions aiming for more efficient and safer transportation at cheaper cost. In the recent years, the technologies of data mining and machine learning have been applied to investigate a range of issues in international freight transportation, supply chain and logistics management, e.g. driver behavior analysis [2], [3], origin-destination parameter estimation [4], pavement maintenance [5], traffic control and forecasting [6], [7], [8], freight logistics [9], air traffic management [10], vehicle classification [11], travel time prediction [12], traffic pattern analysis [13], freight

demand prediction [14], traffic volume forecasting [15], transportation cost forecasting [16], etc. A good literature review regarding utilizing machine learning on freight transportation and logistics applications has been published in [18].

The objective of the FedCSIS 2022 challenge [1]¹, which is in cooperation with PTI and QED Software and sponsored by Control System Software², is to forecast the costs of forwarding contracts, which are, although rather useful in the business providing logistics service or involving supply chains, quite challenging since it can be affected by many static/dynamic and internal/external factors. Besides contract nature and transportation arrangement, the actual transportation cost is constrained by the factors such like fuel prices, currency exchange, drivers behavior, weather, traffic, market demand, etc [16]. There is quite little work published in the literature on cost prediction of forwarding contracts. In [17], AI based models were developed to predict the long-term cost of the logistics service, and attempted to construct a risk-aware interval for the prices to be offered in the bid, aiming to boost competitiveness in the application for tenders. In addition, historical data was used to develop statistical learning models for predicting the success likelihood of a tender based on the actual data and predicted service prices achieved from previous stage. The work proposed in [16] identified the most significant predictive criteria by a trapezoidal neutrosophic fuzzy analytical hierarchy process (TNF-AHP) and based on the criteria found the transportation cost was predicted with an artificial neural network (ANN) model, which, claimed by the authors, can also be employed in supply chain management and inventory control management.

In this paper, an ensemble learning model based on gradient boosting decision trees together with efficient feature engineering and model hyper-parameter optimization has been developed for predicting the costs of forwarding contracts to complete the task given in FedCSIS 2022 challenge. Gradient boosting decision trees (GBDT), developed in the late nineties, is a commonly used boosting methods for solving regression and classification problems in the form of an ensemble of decision trees as weak prediction model, which achieves state-of-the-art results for many commercial and academic applications [19], [20]. In the GBDT, each new model correlates

¹<https://knowledgepit.ai/fedcsis-2022-challenge/>

²<https://controlsystem.com.pl/>

to the negative gradient of the system's loss function that is minimized by using the gradient descent method, which is successively fitted to delivery better estimation of dependent variables via training, resulting in gradual improvement of prediction accuracy. Three efficient GBDT implementations, i.e. XGBoost, CatBoost, and lightGBM, which have shown their powerful learning capabilities by many winning teams in a number of machine learning competitions, are employed to construct the ensemble learning model for forecasting the cost of contract forwarding in our method.

The remainder of the paper is organized as follows. The FedCSIS 2022 Challenge is briefly described in Section II. Data transformation and feature engineering is presented in Section III, followed with the description of the gradient boosting models, model diversity, and ensemble learning in Sections IV, V and VI, respectively. The experimental results in Section VII. Concluding remarks are provided in Section VIII.

II. FEDCSIS 2022 CHALLENGE

The FedCSIS 2022 data mining competition focused on the prediction of the costs of forwarding contracts' execution based on 6 years of detailed history of orders on the European transport exchange. The data contained both the general information about the contracts as well as detailed data of planned routes' segments including geo-located and timed path, specification of shipping vehicles and cargo and even financial details including daily currency rates and wholesale fuel prices. The objective of the competition was to develop a prediction model to accurately estimate the total cost of the contract execution based on all available data. The competitors were provided with the training data from 330055 contracts along with the true realized cost, as well as the testing data from 72452 contracts but without the realized cost. The knowledgepit.ai platform³, on which the competition was hosted operated a leaderboard, which provided the feedback to the competitive model prediction submissions in a form of the preliminary RMSE score⁴ computed over the unknown 10% of the testing set, while the final RMSE score for the complete testing set - constituting the final results, were provided after the submissions' closure.

III. DATA TRANSFORMATION AND FEATURE ENGINEERING

The data provided by the competition organizers included already a well curated, cleaned and carefully selected set of features, however only main dataset providing general contract details was organized in a tabular format of one row (record) per forwarding contract. The extended route data, on the other hand, contained detailed records of between 1 and 31 subsequent steps of the planned route segments of the same contract and hence it became immediately clear that in order to build a competitive cost prediction model all the individual steps data would have to be incorporated hence eventually somewhat aggregated per each contract. We

have developed a generic aggregation filter and applied it all useful columns of the detailed route segments dataset to achieve per-contract aggregates. For numerical columns eleven self-explanative aggregators were applied: 'first', 'last', 'min', 'max', 'argmin', 'argmax', 'mean', 'mode', 'sum', 'range', 'std'. For categorical columns the aggregation treatment was made dependent on the number of unique values. For more than 100 unique values the occurrence of each value was considered sparse enough to limit the aggregation to just the four operators of 'first', 'last', 'mode', 'nuq', where 'nuq' simply denotes the number of unique elements. For categorical columns with fewer than 100 unique values aside of the above-mentioned 4 categorical aggregators we have also applied one-hot-encoding on the original feature and with thereby up to 100 new numerical columns we have applied again all the 11 above-mentioned numerical aggregators to receive quite a large number of final features in the order of thousands. To avoid redundancy and wasteful poor quality features we have automatically eliminated duplicate features and removed features with at most one unique value different than nan/null, which typically resulted in the final set of up to 2000 features.

We have also included features that measured country disagreement between the origin and destination, extracted days of the week, various expected segment duration and the prices of different type of fuel during the trip segment days. Among the alternative but less successful data preprocessing techniques we have explored flattening all trip segments along the single contract record of up to 31 possible segments as well as organizing the data as sequences of consistent route step segments.

IV. GRADIENT BOOSTING MODELS

Preliminary experiments on the main dataset very clearly revealed that gradient boosting models performed by far the best in terms of the reference predictive accuracy and actually quite well in terms of the computational cost, even comparing to simple linear regression and by far comparing to deep networks. Among gradient boosting models XGBoost, LightGBM, CatBoost were used and subsequently optimized throughout the competition. Their variants trained on different parameters were utilized for second level ensembles both executed by simple aggregation and by stacked retrained ensemble of gradient boosting model versions.

A. Individual models' parametric optimization

Current state of the art Machine Learning models are highly customized and flexible to accommodate a very wide range of different options, versions and parametric settings during the model build. Gradient boosting models are good examples of such models with tens of algorithmic, representational, modelling and statistical parameters available to tune in to best fit or represent the data and ultimately to learn robust regression function between the inputs and the continuous output that generalizes well on the previously unseen data.

Given a set of distinct models each with a large numbers of parameters to tune we have decided to apply fast greedy,

³<https://knowledgepit.ai/>

⁴https://en.wikipedia.org/wiki/Root-mean-square_deviation

rotational grid search for each of the gradient boosting models: XGBoost, CatBoost and LightGBM. Each optimizable parameter, whether numerical or categorical is assigned up to 5 unique values comprehensively covering the domain of this parameter. Contrary to the exhaustive parametric grid search, which given the numbers of parameters in our case would prove intractable, our method incrementally finds local optimum of a specific parameter with remaining staying fixed, before rotationally progressing to the next until no improvement can be found from any local change. To further boost the reliability of the best found configurations of parameters we have applied 5-fold cross-validation to rule out accidentally high performance, yet as a consequence to limit an additional cost of cross-validation the process of local optima search for each parameter was reduced to just a pair of neighbouring checks: above and below the current value per turn and shifting the current optimal to the value for which the maximum performance improvement was reported.

This parameters optimization process is terminated when no improvement in cross-validated RMSE performance was found from any local changes of parameters.

V. MODEL DIVERSIFICATION TECHNIQUES

Well performing but diverse models produce diverse outputs which after aggregation produce significant reduction of both variance and bias error components. The critical challenge here is how to develop diverse but well performing models and also to which degree worse performing but quite diverse models are still worth combining to achieve the performance gain. We have developed two generic diversification methods applicable to gradient boosting models. The first method focuses on maximizing the number and magnitude of differences between as many parameters as possible of the same model. The second method takes specific categorical focal feature with a few unique values and proceeds with training an array of models specific to each of the unique value of the focal feature. Both method yield good results with parametric-diversity achieving lower levels of output decorrelation but higher individual performances, while decompositional-diversity achieving higher diversity but lower performance mainly due to smaller number of training examples to train on.

A. Parametric model diversification

Parametric model diversification method was developed in conjunction with the parametric model optimization discussed above. The method simply retains model parametric configurations and the corresponding regression accuracy throughout the optimization process and tries to establish the the population of the best performing model versions with the most diverse configurations of the parameters. To assess the level of diversity among model versions' parametric configurations we developed a simple disagreement measure adding up the differences in grid positions of all parameters conceptually similar to the City Block (Manhattan) distance metric. Once all parameter configurations encountered throughout the optimization are evaluated in terms of their performance P and the diversity

measure D , the final step involves selecting k best model versions from the performance-diversity profile which could be associated with the normalized ratio of $D/RMSE$ assuming our performance measure is $P = 1/RMSE$. Selected models versions outputs are then subsequently aggregated using the simple average operator.

B. Decompositional model diversification

It is known that certain level of model diversity, seen as the level of disagreement among model outputs, could be achieved by the training on mutually exclusive data sets. This effect can be further reinforced if the instead of training on random partitions of the training set, model versions are trained on partitions associated with the different values of certain categorical variables as they typically represent significantly different subsets of the available data. It can be justifiable argued, though, that any limitation or reduction of the training set size exposed to the model is likely to reduce its predictive performance. While it indeed could be the case, particularly for the small training data sets, we argue and have experimentally verified that when the data set is large enough and the categorical variable has only very few unique values, the benefits from combining the outputs from that way diversified model versions outweigh the negligible reduction in performance of a single model trained on the whole set. Given all our models utilize the decision tree construction mechanism in the back end, such guided data partitioning could be considered as a forced first splits that branch out into several categorical-variable-value specific tree-based model versions. Given 300k+ size of the training set we have identified several suitable categorical variables with only a few unique values and trained boosting models on subsets corresponding to specific values of these variables obviously without this variable included in the training process. Trained model versions were then applied separately to the testing sets to generate the outputs which were finally combined with the simple mean operator to produce a single output. Figure 1 comparatively illustrates how the decompositional model diversification differs from the traditional individual model build along the training and testing processes.

VI. ENSEMBLE MODEL

In the construction of the final ensemble, we have utilized 3 baseline gradient boosting models: XGBoost (XGB), LightGBM (LGBM), and CatBoost (CatB) subjected to both parametric (DivP) and decompositional (DivD) diversification filters. Diversification techniques are designed to improve the classifier generalization performance by first expanding it into a number of diverse versions, train them on a whole or subsets of the training set and apply them to the testing set before merging the model versions' outputs back together by a simple aggregation. To further boost diversity but also in a search for better complementary predictive performance we have trained all baseline regression models with their DivP/D filters on two different subsets of features generated by our feature engineering engine. The only difference between these two

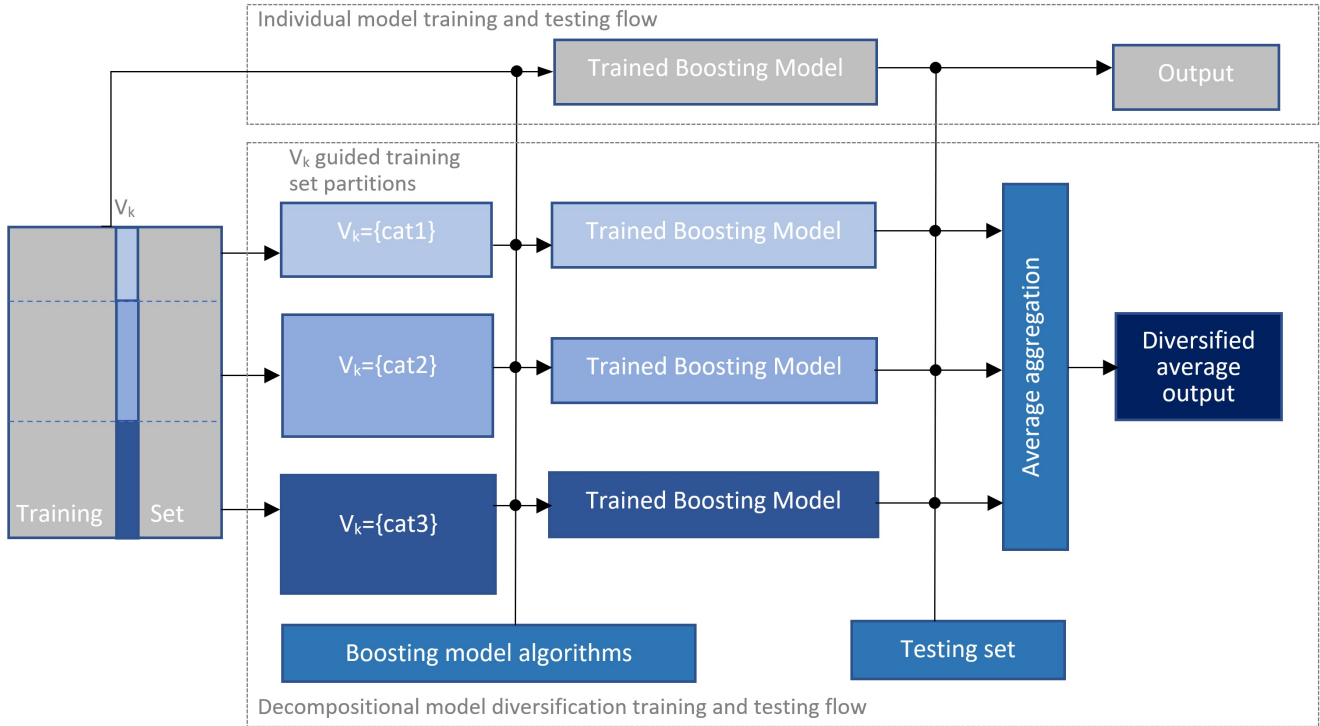


Figure 1. Decompositional model diversification compared to traditional individual model training and testing flow chart.

feature subsets were that the second set included many more sparse columns extracted from much more prolific application of one-hot-encoding to categorical features.

Moreover, in a search for further performance gains we have added another stacked layer of simple linear regression trained on the outputs from diversified baseline models. To properly accommodate stacking layer the training data were split into two parts, one used for building the baseline models and their diversified versions, while the other for learning the parameters of the linear regression in the stacking layer.

Eventually all diversified individual model outputs along with the outputs from linear regression based stacking were averaged together. The architecture or rather flow chart of the final ensemble is depicted in Figure 2.

VII. EXPERIMENTAL RESULTS

To establish a baseline predictability for the presented problem of forwarding cost prediction we first optimized individual gradient boosting models: XGB, LGBM and CatB on two above-mentioned subsets of extracted features and received the following RMSE results along with the optimal parameters returned by the optimization process.

1) Feature set 1 with 894 features:

- CatB (learning rate 0.05, depth 8, iterations 2000): 0.1442
- LGBM (learning rate 0.02, depth 8, iterations 2000): 0.1444
- XGB (learning rate 0.02, depth 1000, iterations 1000): 0.1496

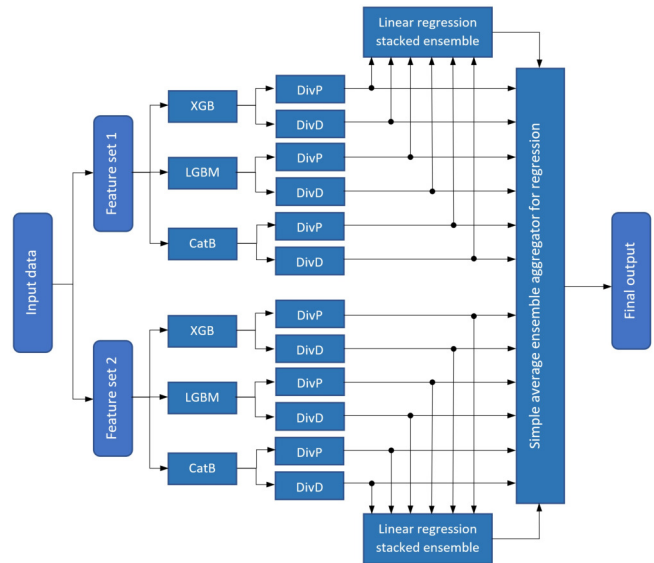


Figure 2. Flowchart of the final ensemble model.

2) Feature set 2 with 3431 features:

- CatB (learning rate 0.05, depth 8, iterations 2000): 0.1513
- LGBM (learning rate 0.02, depth 8, iterations 2000): 0.1494
- XGB (learning rate 0.02, depth 1000, iterations 1000): 0.1640

Secondly, linear regression stacking models were trained on the outputs from the diversified individual models and reported the preliminary RMSE performance of:

- Stacking model with 894 features: 0.1441
- Stacking model with 3431 features: 0.1465

The final predictions are achieved by ensemble averaging of the outcomes from both stacking models along with diversified individual baseline models, yielding the RMSE around 0.141.

A. Visual efficient performance-diversity horizon method for a robust ensemble composition

Even though the ensemble model illustrated in Figure 2 resulted with the best preliminary performance, the journey of incremental model build and performance improvement led to hundreds of model output submissions as potential solutions each with different performance and mutual dependency with the other solutions. Given the submissions represented the outputs from multiple different models with many different versions of the training data, parametric setups and model design choices we have considered them as a final-stage resource for potential further performance improvement through simple aggregation. The question posed in this stage was: given n model outputs with preliminary RMSE scores is it possible to improve the best model result and if so how to achieve the biggest possible improvement.

The intuition backed by the bias-variance error decomposition theorem suggested that the best results should be achieved by combining the outputs from the best performing (model) solutions that differ the most from each other, inline with the diversification techniques discussed in section V. At the final stage when only model outputs and their preliminary scores were available the natural disagreement or resultant diversity measure that could be computed upon the model outputs was an average from the correlation coefficients between the specific model output and all the rest in the considered pool of solutions. After computing such outputs' diversity c_i for all solutions with the preliminary RMSE scores $e_i < 0.152$ we have plotted all such best solutions from our submissions as points (c_i, e_i) on the 2-dimensional diagram depicting dependency $e = f(c)$, as shown in Figure 3.

The points stretching along the bottom horizon approximately marked by the dashed line represent the continuum of the best performing and at the same time the most diverse solutions and are expected to be the most promising choices for final stage combination to achieve the performance improvement. Rather than arbitrarily take some solution for combination along such horizon we have developed a simple greedy algorithm to choose the best solution candidates from around the the horizon for final combination. The greedy sequence in our case is starting from the best performing model in the bottom right corner and then adding the next best model but only out of the more diverse solutions, i.e. from the top solution as a pivot the next solution added is represented by the lowest point to the left from the current pivot. Then the pivot is shifted to the newly added point and process repeats until no more points can be added. Such greedy sequential

addition leads to the staircase connected set of points marked in black along the diversity horizon. The final stage is testing at what point such greedy sequence does not improve the overall ensemble performance any more.

Such method of output-level ensemble combination is particularly effective when a large number of black-box models are suddenly at the disposal and a quick decision is needed on which models' outputs are best to aggregate to maximally reduce the overall predictive error. In our cases the team merger pose an exact situation as described above and following a quick testing a final ensemble output has been generated by aggregation of the outputs from the first 11 models along the diversity horizon, and yielded the top predictive RMSE error below 0.14 and even better result on the full testing set, thereby securing the first place in the FedCSIS 2022 competition.

VIII. CONCLUSIONS

We have attempted to improve predictive performance of the already highly robust regression models from the gradient boosting family: XGBoost, LGBM, CatBoost. To achieve that we have proposed a range of model diversification methods coupled with various ensemble combination schemes. Compositional diversity forced by training on significantly different input data subsets, combined with actively encouraged parametric diversity led to an improvement in performance achieved from aggregation of the expanded diverse model versions, additionally boosted with linear regression based stacking and output level selection of the most efficient ensemble candidates in terms of the performance-diversity trade-off. The proposed ensemble has been applied to the complex problem of advance prediction of the total realized cost of forwarding contracts' based on a variety of data coming in different forms and types, in the competitive setup of the FedCSIS 2022 data mining challenge. Our proposed solution scored the first place in this challenge producing the lowest (RMSE) error below 0.14, which corresponds to only 2% in relative cost in monetary units. The proposed solution can enable forward contractors to better estimate their expected shipment cost, further reducing their business risks and boosting the efficiency across transport services domain.

REFERENCES

- [1] A. Janusz, A. Jamiołkowski, M. Okulewicz, Predicting the Costs of Forwarding Contracts: Analysis of Data Mining Competition Results, *Proceedings of the 17th Conference on Computer Science and Intelligent Systems (FedCSIS)*, 2022.
- [2] Z. Li, K. Zhang, B. Chen, Y. Dong and L. Zhang, Driver identification in intelligent vehicle systems using machine learning algorithms, *IET Intell. Transp. Syst.*, vol. 13, no. 1, pp. 40-47, 2019.
- [3] S. Bhattacharya. Novel approach for Ai based driver behavior analysis model using visual and cognitive data, 2019.
- [4] S. Kikuchi, R. Nanda and V. Perincherry. A method to estimate trip O-D patterns using a neural network approach. *Transp. Planning Technol.* 17(1): 51-65, 1993.
- [5] A. Pozarycki. Pavement diagnosis accuracy with controlled application of artificial neural network, *The Baltic Journal of Road and Bridge Engineering* 10(4): 355-364, 2015.
- [6] M. Bielli, G. Ambrosino, M. Boero and M. Mastretta. Artificial intelligence techniques for urban traffic control. *Transportation Research Part A: General* 25(5):319-325, 1991.

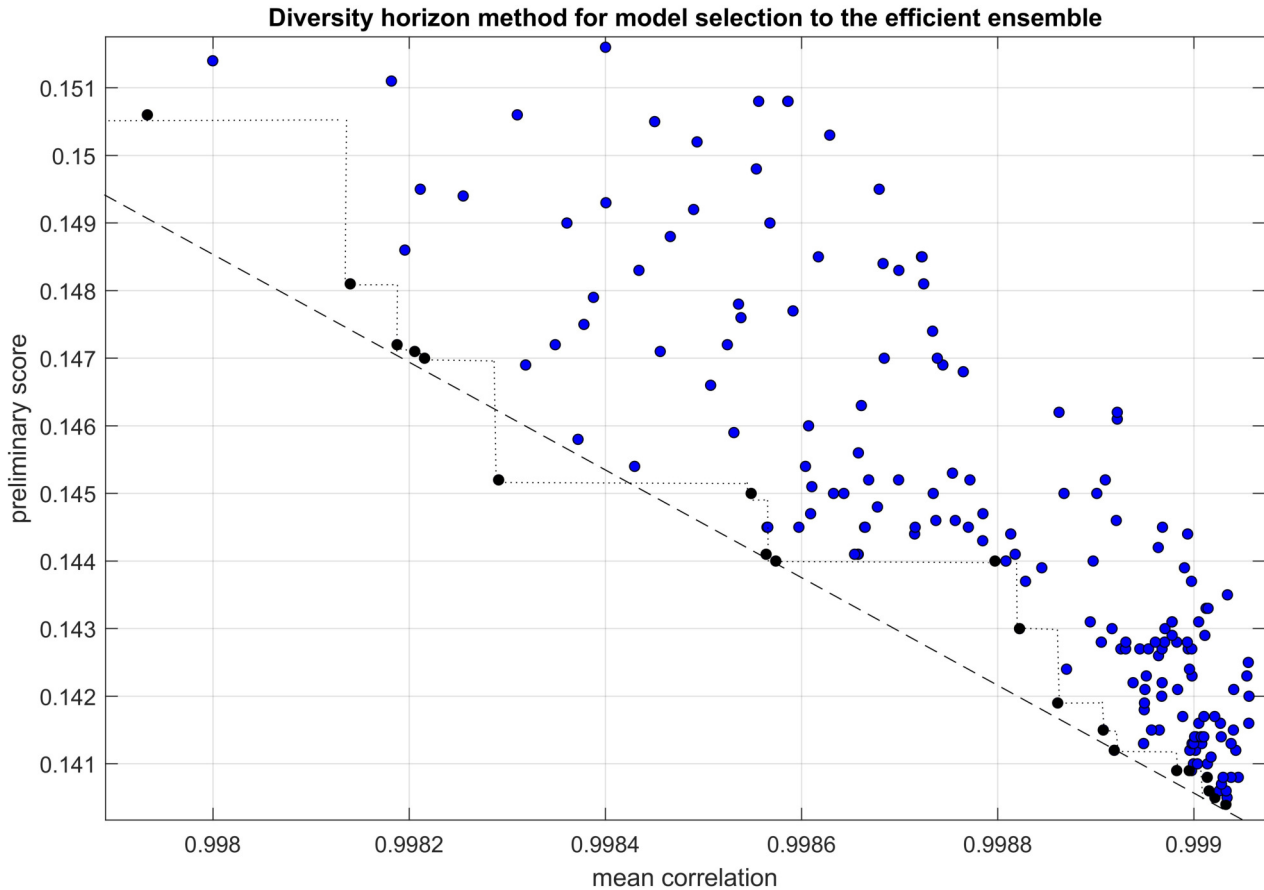


Figure 3. Diversity horizon method for model selections.

- [7] M. Ghanin and G. Abu-Lebdeh. Projected state-wide traffic forecast parameters using artificial neural networks. *IET Intel. Transp. Syst.* 13(4):661-669, 2019.
- [8] J. Lu, L. Feng, J. Yang, M. Hassan, A. Alelaiwi and I. Humar, Artificial agent: The fusion of artificial intelligence and a mobile agent for energy-efficient traffic control in wireless sensor networks, *Future Gener. Comput. Syst.* 95:45-51, 2019.
- [9] Y. Kayikci. A conceptual model for intermodal freight logistics centre location decisions. *Proc. Soc. Behav. Sci.* 2(3): 6297-6311, 2010.
- [10] F. Saadaoui, H. Saadaoui and H. Rabbouch. Hybrid feedforward ANN with NLS-based regression curve fitting for US air traffic forecasting. *Neural Computing and Appl.* 32:10073-10085, 2019.
- [11] J. George, A. Cyril, B. Koshy and L. Mary. Exploring sound signature for vehicle detection and classification using ANN. *Int. J. Soft Comput.* 4(2):29-36, 2013.
- [12] H. Lin, R. Zito and M. Taylor. A review of travel-time prediction in transport and logistics. *Proc. Eastern Asia Soc. Transp. Stud.* 5:1433-1448, 2005.
- [13] H. Kirby and G. Parker. The development of traffic and transport applications of artificial intelligence: An overview. *Artificial Intelligence Applications to Traffic Engineering*, The Netherlands:VSP, pp. 3-27, 1994
- [14] I.C. Bilegan, T.G. Crainic and M. Gendreau. Forecasting freight demand at intermodal terminals using neural networks—an integrated framework. *Eur. J. Oper. Res* 13(1):22-36, 2008.
- [15] K. Kumar, M. Parida and V. Katiyar. Short term traffic flow prediction for a non urban highway using artificial neural network. *Proc. Social Behav. Sci.* 104:755-764, 2013.
- [16] A. Singh, A. Das, U.K. Bera and G.M. Lee. Prediction of Transportation Costs Using Trapezoidal Neutrosophic Fuzzy Analytic Hierarchy Process and Artificial Neural Networks. *IEEE Access* 9:103497-103512, 2021.
- [17] S. Nataraj, C. Alvareza, L. Sadaa, A. Juana, J. Panaderoa and C. Bayliss. Applying Statistical Learning Methods for Forecasting Prices and Enhancing the Probability of Success in Logistics Tenders. *Transportation Research Procedia (Elsevier)* 47:529-536, 2020.
- [18] K. Tsolaki, T. Vafeiadis, A.N. Dimosthenis and I.D. Tzovaras. Utilizing machine learning on freight transportation and logistics applications: A review. *ICT Express*, 2022.
- [19] L. Mason, J. Baxter, P.L. Bartlett, and M. Frean. Boosting Algorithms as Gradient Descent In S.A. Solla and T.K. Leen and K. Müller. *Advances in Neural Information Processing Systems* 12: 512-518, MIT Press, 1999.
- [20] J.H. Friedman. Greedy function approximation: A gradient boosting machine. *Ann. Stat.* 29(5): 1189-1232, 2001.

Application of Diversified Ensemble Learning in Real-life Business Problems: The Case of Predicting Costs of Forwarding Contracts

Milena Trajanoska^{*}, Pavel Gjorgovski^{*}, Eftim Zdravevski

Faculty of Computer Science and Engineering

Ss.Cyril and Methodius University, Skopje, Macedonia

ORCID: {0000-0003-0105-7693, 0000-0002-6859-4402, 0000-0001-7664-0168}

Abstract—Finding an optimal machine learning model that can be applied to a business problem is a complex challenge that needs to provide a balance between multiple requirements, including a high predictive performance of the model, continuous learning and deployment, and explainability of the predictions. The topic of the FedCSIS 2022 Challenge: ‘Predicting the Costs of Forwarding Contracts’ is related to the challenges logistics and transportation companies are facing. To tackle these challenges, we established an entire Machine Learning framework which includes domain-specific feature engineering and enrichment, generic feature transformation and extraction, model hyperparameter tuning, and creating ensembles of traditional and deep learning models. Our contributions additionally include an analysis of the types of models which are suitable for the case of predicting a multi-modal continuous target variable, as well as an explainable analysis of the features which have the largest impact on predicting the value of these costs. We further show that ensembles created by combining multiple different models trained with different algorithms can improve the performance on unseen data. In this particular dataset, the experiments showed that such a combination improves the score by 3% compared to the best performing individual model.

Index Terms—Costs of Forwarding Contract, explainability, prediction ensembles, Diversified Ensemble Learning

I. INTRODUCTION

TO BE competitive in the market, companies need to be able to utilize all available data and perform analytics to identify hidden patterns [1]. This can allow them to improve their processes, better understand their customers, and make predictions (e.g. churn prediction, service-outage prediction, fraud detection, etc.). To achieve such goals, companies are facing a variety of challenges, ranging from data integration from a variety of sources [1] and finding suitable machine learning models that are both performant but also practical and explainable [2], to maintaining the corresponding infrastructure. To perform such analytical data processing and machine learning on a large scale, companies require a complex computing infrastructure and methods that will minimize their

total cost of ownership [3], and yet scale the computation to multiple nodes [4].

In this paper, we focus on the problem of finding an optimal machine learning algorithm that can be easily applied in a real-life business domain, meaning it should achieve high predictive performance, continuous learning and deployment, and explainability of the models and their predictions. The topic of the FedCSIS 2022 Challenge, hosted on the KnowledgePit portal is ‘Predicting the Costs of Forwarding Contracts’ [5]. The competition addresses the challenges of transportation, shipping, and logistics companies related to their digital transformation. Particularly, the benefits of the research boosted by this competition for such companies can be multi-fold:

- Identify reasons and circumstances that lead to increased transportation costs.
- Improve companies’ planning to lower the costs, and generally, improve their investment strategy.
- Help companies in selecting contracts that maximize their profits by predicting the forwarding (i.e., delivery) contract cost.

Similar real-world challenges were addressed at previous competitions on the KnowledgePit platform, such as predicting escalations in customer support [6], network device workload prediction [7], suspicious network event recognition [8], and predicting victories in video games [9], to name a few. These papers also demonstrate how predictions from individual solutions could be integrated into diversified ensembles to create more powerful and more robust models.

For the case study on which we focus in this paper, we choose to use the XGBoost model with a grid search with 5-fold cross-validation [10], due to its extensive use in retail sale predictions [11]. We also use Random Forest models with grid search, as well as deep learning models that are commonly used in demand forecasting in multi-channel retail [12]. Finally, the Linear Regression models are one of the most commonly used simple models for price prediction in industry

^{*}These authors contributed equally to this work.

settings [13]. All hyper-parameter tuning is done via exhaustively searching a specified subset of the hyper-parameter space of the given model. The validation is performed using a 5-fold cross-validation, which balances validation speed and metric accuracy of the test data.

The rest of the paper is structured as follows. Section II reviews the most important related works. Section III describes the experimental setup and multiple validation procedures we used to evaluate each model we have developed in this study. Section IV describes the preprocessing of the data, including data transformations and aggregations. Subsection IV-A contains information on the preprocessing implemented over the main table training and testing data, and subsection IV-B includes the preprocessing information for the routes table training and testing data. The section V describes the implemented feature selection methods. The experiments, including model hyper-parameter tuning, training, and evaluation techniques are described in section VI, along with an overview of the final scores of the implemented models. The paper concludes with section VIII where we give a brief overview of the entire Machine Learning workflow, limitations to the study, and opportunities for further work and improvement of the methods.

II. LITERATURE REVIEW

Even though similar challenges have been extensively studied in other industries, this problem is fairly new in the logistics sector. Authors of [14] analyze the shipping cost differences between various carriers, and attempt to identify opportunities for reducing transportation costs.

Similarly, in [15], authors utilize neural networks to forecast shipping freight rates and compare them with traditional time series analysis models. The key objective of their work is to improve the forecasting accuracy of traditional time series analysis. In relation to the competition task, this article also highlights the importance of the information contained in forwarding freight agreements in relation to predictive accuracy.

Another interesting approach is presented in [16], where the impact of the demand and cargo capacity on the shipping price is identified. Forecasting of the long-term cost of logistics contracts is particularly important in long-term agreements with upfront-defined prices, such as in various types of tenders and auctions. On one hand, the bids should be attractive so that the contract can be won, while still being profitable for the logistic company. This challenge was researched in [17], which utilized historic data to train the models.

On a related topic, Men et al. [18] use an ensemble of mixture density neural networks for the purpose of short-term wind speed and power forecasting. They show that this methodology works well for multi-step ahead prediction. Additionally, [19] illustrates the use-case of multi-observation and multi-dimensional data cleaning methods for applying machine learning algorithms. In this study [19], the authors use transactions from the Lending club data set for training tree-based models to predict peer-to-peer (P2P) loan default and observe that the LightGBM algorithm, using multiple

observational data, has the best performance. In many cases, it has been shown that decision tree-based methods significantly outperform linear models for predicting complex response variables, such as the example of predicting accrual expenses in a balance sheet by utilizing the unused vacation time of employees [20].

The scientific community has placed a massive effort into studying individual algorithms (e.g. ensemble algorithms, various deep learning architectures, etc.). Additionally, some studies also focus on finding ways to utilize the diverse algorithms and integrate their predictions. This process is often referred to as diversified ensemble learning and aims to find the best classification algorithms (out of many heterogeneous classification algorithms) and an optimal method to combine them [2]. Note that the individual algorithms used in a diversified ensemble could be ensembles on their own (e.g., XGBoost [21] or Random Forest [22]), so the term diversified ensemble learning refers to another layer of integration. Some methods train another classifier whose inputs are the predictions of the individual classifiers [23] or use other ways of voting. In this paper, algorithms perform weighted voting based on empirically identified weights.

III. VALIDATION PROCEDURE

As in all practical machine learning problems, the experimental setup concerning the training/validation/test split should resemble the natural chronological and logical process as closely as possible, so that the models built are valid and robust over time. In that regard, we attempted to split the training dataset into two subsets, one for training and one for validation, in a way that we thought would most resemble the natural setting in which the data was collected. Considering that this is a very practical problem coming from the industry, any results of the transformation and validation methods should be applicable in a production setting.

That being said, we considered the *id_payer* column, the client identifier, as special because it gave us the ability to use it primarily for splitting the original training set into our training and validation subsets. For this purpose, we first analyzed the frequency of rows in the main table per *id_payer*, dubbing it *number_of_contracts*. We noticed the huge discrepancy in the frequency of contracts, ranging from just a few to upwards of thousands. Therefore, we tried several approaches in how we considered this fact:

- **Split by alternating frequency of records per *id_payer*.** In this approach, we ordered the *id_payer* records by the *number_of_contracts*, and we assigned them to our training or validation split in alternating order. The idea was that roughly 50% of the records will be our training set, and the other 50% will be the validation set. One additional benefit of this approach was that it made sure that the *id_payer* column would not have an effect on the prediction. With this approach we are very conservative to overfitting, trying to train the models on one subset of the data, and applying the models to a completely new set of data. Indeed, our

first submissions showed that our own validation results were considerably worse than the leaderboard results, but were still consistent when comparing different algorithms or feature subsets (the better models per our internal evaluation were also better on the leaderboard).

- **Time-sensitive split.** We also tested splitting the data in such a way that the older contracts (records with an earlier start date) were in the training set, while newer records were in the validation set. This approach mitigates the previous conservativeness, by allowing the same clients to be in the training and validation set, while also allowing some new clients to appear in the validation set.

After the initial testing of the previous approaches, we noticed that the hyper-parameter tuning procedures performed on our hold-out training set (a subset of the competition training set) were not fully applicable when we used the whole training dataset provided in the competition. Namely we used our hold-out training set and the remaining of the training set to learn the hyper-parameters. Then, we compared two models trained with the same hyper-parameters – one using the hold-out training set, and another trained on the full training dataset. The former performed significantly better on the leaderboard result, even though it was trained on smaller data set. With this counter-intuitive finding that contradicts the common principle that more training data is better, and having a very limited time for this competition, we decided to use 5-fold cross-validation in the remaining experiments so that we can use the full training dataset for making the final test predictions. Despite that, we strongly believe that further experiments in the validation procedure are needed to properly tackle the problem.

IV. DATA PREPROCESSING

After the initial data exploration phase, we decided to primarily focus on the main table and extract whichever knowledge we can from it, before proceeding with utilizing the detailed table of expected routes.

A. Main Table

Firstly, the *prim_train_line* and *prim_ferry_line* features were not used, due to the high missing data ratio (between 80% and 90% missing from the total number of observations). Additionally, these columns had unstandardized data (e.g., temperature ranges or temperature and unit combined strings in the same column as a descriptive field, etc.) For the remaining columns which had missing data, we applied mean (for continuous columns) or median filling (for nominal data).

The transformations done on the Main Table were split into two major types:

- One-hot encoding of categorical (nominal) data. We considered utilizing the Weight of Evidence [24] approach, but considering that the categorical features had a relatively small number of different values, the one-hot encoding technique was considered sufficient.

- Combinations of two or more features to create a new meaningful feature. Such features were a result of calculations based on the columns that contain date or timestamp information.

1) *Nominal to numeric features with one-hot encoding:* The one-hot encoding was done to maximize data balance while minimizing the loss of information. Binary features or features with a few different values were transformed with classic one-hot encoding. The features with over 15 values were split into 3 major categories: low-frequency categories (those that had appeared under 1000 times in the data set), high-frequency categories (those that had appeared over 1000 times in the data set), and the highest frequency category of the feature was separated as an individual category.

2) *New domain-specific features based on other features:*

a) *Date-time related features:* A combining of two or more features was done for the *route_start_numeric* and *time_taken_minutes* features. The *route_start_numeric* is the difference in days between the minimum date found in the dataset (i.e., 1/1/2016), and the start date of the specific route. This was done by using the *route_start_datetime* feature, and finding the number of days between it and 1/1/2016. Similarly, the *time_taken_minutes* is the time the complete route is estimated to take in minutes. This is calculated by finding the difference in minutes between the *route_start_datetime* and *route_end_datetime* features.

b) *Geo-spatial features:* To enhance the geo-spatial information about the routes, we created a new feature, using the Euclidean distance [25] between the *route_starting_point* and *route_ending_point*. This calculation uses the latitude and longitude values of the original points. Additionally, we used the geo-spatial (Haversine) distance [26], given in the competition dataset.

B. Routes table

The initial experiments were conducted using only data from the Main Table. To further improve model performance in later experiments, we enriched the dataset with aggregate features extracted from the Routes Table.

The columns which had missing data were very sparse in the general case. Moreover, the lack of entries seemed correlated in most cases. For this reason, we decided to ignore such columns and do not create features based on them, especially considering the limited time we had for experiments. Still, we believe that more sophisticated data imputation methods could be explored in the future, or at least to prepare some bins of values in cases when such data is available. The ignored columns for this reason were: *ferry_line*, *train_line*, and another 17 columns whose names started with *vehicle_* or *id_vehicle_*.

We have extracted the aggregate features by grouping the dataset based on the column *id_contract*. Before the aggregation, one-hot encoding was performed on the *step_type* feature with the goal of extracting the number of steps of each type that were taken in one route. The following features were extracted for each route:

- *num_steps_with_vehicle* - the number of rows having *id_vehicle* equal to 1
- *num_steps_with_trailer* - the number of rows having *id_trailer* equal to 1
- *num_steps* - the maximum from the *step* column values within one *id_contract* partition
- *num_steps_A* - the number of steps of type A
- *num_steps_B* - the number of steps of type B
- *num_steps_D* - the number of steps of type D
- *num_steps_F* - the number of steps of type F
- *num_steps_K* - the number of steps of type K
- *num_steps_N* - the number of steps of type N
- *num_steps_O* - the number of steps of type O
- *num_steps_P* - the number of steps of type P
- *num_steps_R* - the number of steps of type R
- *num_steps_S* - the number of steps of type S
- *num_steps_W* - the number of steps of type W
- *num_steps_Z* - the number of steps of type Z
- *num_steps_A* - the number of steps of type A
- *num_steps_empty* - number of the steps in which the *if_empty* flag was equal to 1
- *num_external_steps* - the number of the steps in which the flag *external_fleet* was 1
- *max_loaded_kg* - the maximum *kg_load_unload*
- *max_unloaded_kg* - the minimum of *kg_load_unload*
- *average_load_step* - the mean of the feature *kg_current*
- *max_load_step* - the maximum of the feature *kg_current*
- *min_load_step* - the minimum of the feature *kg_current*
- *num_steps_ferry* - the number of steps in which the flag *ferry* was equal to 1
- *num_steps_train* - the number of steps in which the flag *train* was equal to 1
- *total_km_train* - the sum of *train_km* for each step
- *max_time* - the maximum of the feature *estimated_time* in each step
- *min_time* - the minimum of the feature *estimated_time* in each step
- *km_per_step* - the mean value of the feature *km* in each step
- *km_nonempty_max* - the maximum value of the feature *km_nonempty*
- *km_nonempty_total* - the sum of *km_nonempty* in each step
- *average_time_per_step_minutes* - the average of the time difference in minutes for each step

V. FEATURE SELECTION METHODS

A. Manual Filtering of Correlated Features

This method was used for training the Linear Regression models. Since the Maximum likelihood (MLE) estimations [27] can be highly disturbed by correlated features, we decided to manually remove features with correlations greater than 0.6 in absolute value. For this purpose, we calculated the Pearson

Correlation [28] between each pair of continuous variables in the main dataset. We only calculated the correlations between the continuous features from the main dataset in order to exclude the features that have a high correlation from the feature engineering step for training the linear regression models.

The linear regression models were only trained using the uncorrelated features from the main dataset, plus higher degrees of some of the most important features chosen by applying domain knowledge. These features include the total kilometers, the time taken and the maximum weight. This was done in order to use the results of the most basic linear regression models as an internal evaluation baseline for all the other trained models.

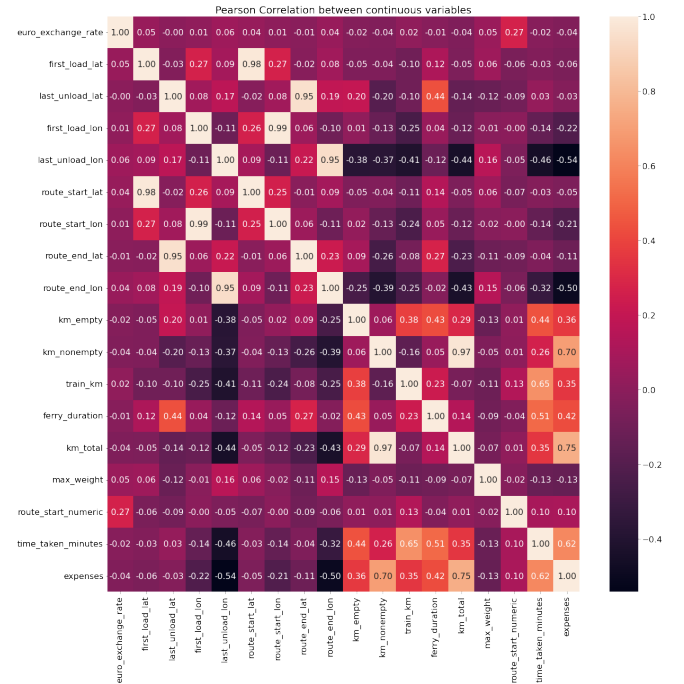


Fig. 1. **Main table extracted continuous features correlation.** This figure presents the Pearson Correlation coefficients calculated between each pair of continuous features from the main table.

The experiments for the Linear Regression models were conducted using only the Main Table data. The correlation map of the continuous features is displayed in Figure 1. As it is evident from the figure, a lot of features have high correlations (positive and negative), so removing these dependencies was one of the feature selection methods we implemented.

B. XGBoost Feature Importance

Another method for feature selection was using the built-in feature importance metric from the Extreme Gradient Boosting model (XGBoost) for regression. The process of optimizing this model is detailed in section VI.

Figure 2 represents the feature importance obtained from XGBoost on the full dataset (containing the main table and routes table data). As we can see from the figure, only one

feature, namely *direction_d* (a binary feature stating whether the direction is d or not) dominates the rest of the features in the dataset.

Sometimes, these built-in feature importance estimates can be inaccurate, as we suspected in this case. The reason behind this is that XGBoost weights the features based on the frequency of splitting, gain and coverage metrics. In the case of categorical variables, the frequency of splitting can be very low, since there are few possible split points, contributing to an overall lower importance than the actual one for categorical variables. Additionally the gain for continuous features can be lower than that of the categorical ones since more split points are possible to be made which in turn can rule out fewer examples in each split compared to the categorical features.

In our case we are creating shallow individual decision trees as weak learners. This means that we have fewer levels in each tree, thus splitting by a binary (or categorical) feature, which is correlated with the target variable, can have a higher value for the gain compared to a continuous feature which is correlated with the target variable because the continuous feature might rule out fewer examples in each split. In turn, this can result in inaccurate calculations of the overall feature importance when using a mix of categorical and continuous features.

For this reason, we further try to extract the important features using wrapper methods [29] over the XGBoost algorithm, which is explained in the following subsection.

C. Boruta search with Shapley values

Boruta search [30] is a wrapper algorithm originally built over the Random Forest classification model, but further extended for all types of decision tree-based models and regression. The method implements feature selection by creating copies of the original features and shuffling them to remove any correlation with the target variable. These features are called shadow features. The algorithm then compares the shadow features' Z-scores [31] to the original features' Z-scores. Each feature that fails a two-sided test for significant difference of importance with the shadow feature with maximum importance is removed from the dataset. The feature importance, in this case, was measured using Shapley values [32]. The Shapley values are often used as a method for explainable AI (XAI) because they reveal the average marginal contribution of a feature value across all possible coalitions.

The results of implementing this feature selection method over the XGBoost algorithm are shown in Figure 3. A total of 48 features were identified as important, and their importance compared to the shadow features is displayed in the figure.

From the figure, we can again see that the features *euclidean_distance*, *direction_d*, and *km_total* dominate in their importance for the algorithm compared to all of the other features in the dataset, which means that their impact is most significant in determining the *expenses* variable's values.

VI. RESULTS

A. Linear Regression

The Linear Regression models were first experimented with by using the features in the main table and standardizing them according to the needs of the algorithm. The *ferry_intervals*, *train_intervals*, and *id_service_type* features were scaled using the Min-Max scaling, while all other features that were continuous were scaled using the Standard scaling [33]. The root mean squared error (RMSE) of this primary model, using the train/validation split for validation, was 0.6703. The RMSE of this model on the leaderboard was 0.4598.

The same model was later modified to include only the features that are not correlated, according to the OLS [34] statistical test for feature relationships and the correlations represented in Figure 1. Using only those features, the model had a RMSE of 0.6942 on the validation data set, and a RMSE of 0.5027 on the leaderboard.

Finally, the squared values of the features *km_total*, *max_weight* and *time_taken_minutes* were added to the model. This improved the model's RMSE on the validation set to 0.5713, and the RMSE of the test set to 0.4309. The coefficients and their significance are shown in Figure 4.

We can see that all of the features have significant coefficients according to the reported p-value. In this case, the total number of features for training the model was 15. Since linear regression might not estimate the coefficients right in the case of a large number of features, we decided to further go with other non-linear models that better handle a large number of features.

Moreover, we examined that the target variable is multimodal, meaning that any model which expects a Gaussian distribution of the target variable will not be suited well in this scenario. For this reason, we mainly focus on tree-based models and ensembles. We further try Gaussian Mixture distributions with neural networks, but due to the time limit, we did not have the resources to optimize these types of models.

B. Extreme Gradient Boost Regression

The first XGB Regressor model that was built only on the main table dataset included *alpha_booster*, *eta*, *lambda*, and *max_depth* as hyper-parameters in the tuning job, using Bayesian Search [35] to find the optimal values. This resulted in an average of 0.34 RMSE on the 5-fold cross-validation, and a 0.1735 RMSE on the leaderboard.

The model was later improved with the addition of the routes table, as well as the selected features from the Boruta Shap search, which resulted in improving the RMSE of the CV to approximately 0.18 and 0.15, respectively on the 5-fold cross-validation and improving the test RMSE on the leaderboard to 0.1649 and 0.1622 respectively.

C. Random Forest

The first Random Forest model was built on the transformed features of the main table. Hyperparameter optimization was done on the *max_depth*, *min_samples_leaf* and

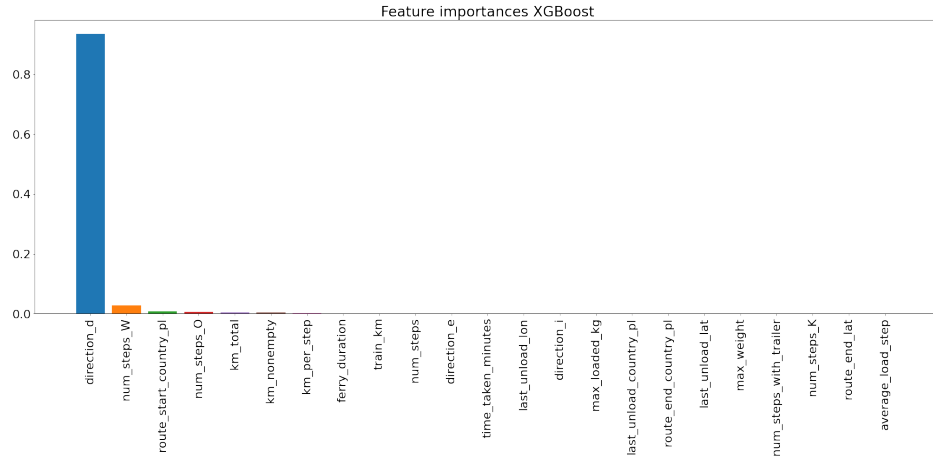


Fig. 2. **XGBoost built-in feature importance for the merged dataset.** This figure represents the feature importance of the main table and the routes table. The horizontal axis represents the features, while the vertical axis represents their corresponding impact on the prediction of the target variable. Only features having an importance greater than 10^{-5} are shown.

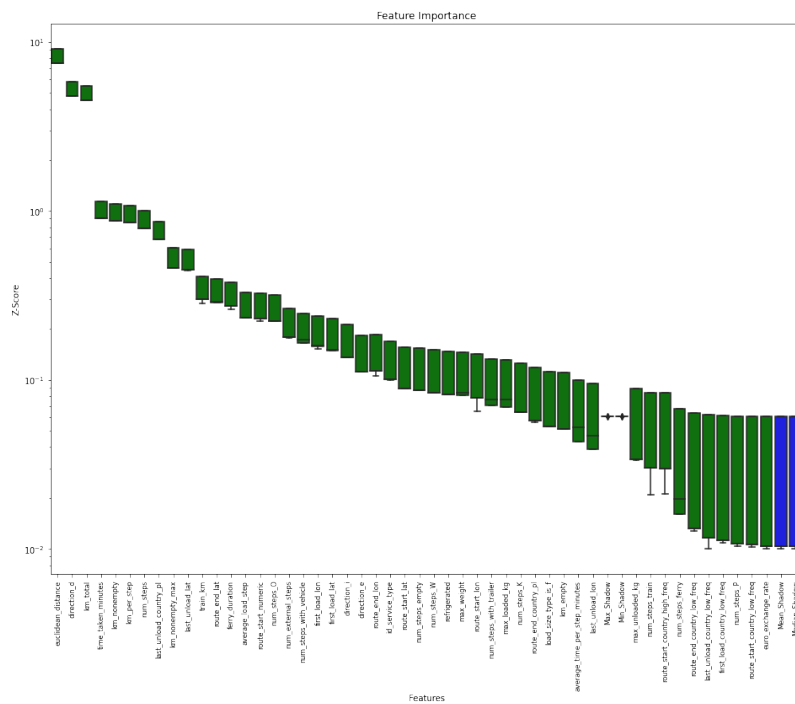


Fig. 3. **Extracted features using Boruta search with Shapley values as an evaluation metric.** The green rectangles represent the original features, while the blue rectangles represent the min, max, mean, and median shadow features. The vertical axis represents the Z-score for each feature.

max_features hyper-parameters. All Random Forest models in this competition had *n_estimators* set to 200. This resulted in a RMSE of 0.0476 on the 5-fold cross-validation, and a RMSE of 0.1726 on the test data.

The model was later improved by using a deeper grid search on all features from the merged main and route tables. This time, *max_depth*, *max_features*, *min_samples_leaf* and *min_samples_split* were optimized using 5-fold cross-validation, which resulted in the validation RMSE being 0.0234. The test results had a RMSE of 0.1625.

Both Random Forest models were clearly over-fitted, however, we tackled that issue with the different Ensembles of models later on.

D. Deep Learning Models

A few feed-forward neural networks were implemented using different configurations. The neural networks were trained on the full dataset. The networks only included dense layers and the main activation function used in the hidden layers was ReLU [36]. We experimented with a few regularization

OLS Regression Results						
Dep. Variable:	expenses	R-squared:	0.870			
Model:	OLS	Adj. R-squared:	0.870			
Method:	Least Squares	F-statistic:	8.659e+04			
Date:	Sat, 14 May 2022	Prob (F-statistic):	0.00			
Time:	16:54:51	Log-Likelihood:	-52431.			
No. Observations:	181108	AIC:	1.049e+05			
Df Residuals:	181093	BIC:	1.050e+05			
Df Model:	14					
Covariance Type: nonrobust						
	coef	std err	t	P> t	[0.025	0.975]
euro_exchange_rate	0.6569	0.005	123.488	0.000	0.646	0.667
first_load_lat	0.0252	0.000	74.723	0.000	0.025	0.026
first_load_lon	-0.0048	0.000	-26.095	0.000	-0.005	-0.004
route_end_lat	0.0245	0.000	78.022	0.000	0.024	0.025
last_unload_lon	-0.0122	0.000	-77.225	0.000	-0.013	-0.012
km_empty	-0.0008	8.15e-06	-95.747	0.000	-0.001	-0.001
km_total	0.0024	4.34e-06	561.506	0.000	0.002	0.002
train_km	0.0008	3.69e-06	220.801	0.000	0.001	0.001
ferry_duration	0.0004	2.39e-06	174.749	0.000	0.000	0.000
max_weight	1.491e-06	1.36e-07	10.976	0.000	1.22e-06	1.76e-06
route_start_numeric	5.525e-05	1.57e-06	35.153	0.000	5.22e-05	5.83e-05
time_taken_minutes	8.148e-07	3.81e-07	2.136	0.033	6.71e-08	1.56e-06
km_total_2	-5.385e-07	1.73e-09	-311.493	0.000	-5.42e-07	-5.35e-07
max_weight_2	-8.619e-12	2.57e-12	-3.358	0.001	-1.36e-11	-3.59e-12
time_taken_minutes_2	-6.906e-11	4.76e-12	-14.522	0.000	-7.84e-11	-5.97e-11

Fig. 4. Coefficients and significance of manually selected features for the linear regression algorithms including squared features. This figure represents the coefficients of the variables included in the linear regression estimation, along with confidence intervals and p-values. All features have significant coefficients.

techniques including dropout, batch normalization, and kernel regularization with L2 [37].

From the conducted experiments, we concluded that adding regularization caused the model to underfit the training data. Moreover, batch normalization caused a significant performance degradation in this case.

For the output layer, we tried two activation functions, namely softplus [38] and linear. In this case, the same network configuration had better performance using the linear activation instead of softplus.

The model which obtained the best results had the following configuration:

- Dense(128, activation=relu)
- Dense(64, activation=relu)
- Dense(64, activation=relu)
- Dense(1, activation=linear)

The results of this model were RMSE of 0.1683 on the random validation split of 20% training data and RMSE of 0.1775 on the leaderboard, respectively.

Adding more layers results in better model performance, however, it also causes the model to overfit the data in the early stages of training.

Since the neural networks approximate a Gaussian-like distribution, they are not quite suitable for the multimodal target in this case. We additionally tried Mixture Distribution

networks but did not have the resources to optimize these models. A basic model with the following configuration of two dense layers with 100 neurons and ReLU activation each for approximating the parameters of the distribution, resulted in a RMSE of 0.1868 on the leaderboard.

E. Diversified Ensemble Models

Ensemble methods [39] were used in order to compensate for models which might be overfitted or underfitted in the data, and further use the errors the models make in a way to further tune the final predictions.

One of the ensemble methods used was model stacking. For this type of ensemble, we used the best performing XGBoost model using the features extracted with the Boruta search method and additionally re-trained a Bagging model of 100 linear regressions with the same features.

The outputs from these models were then fed to another linear regression model, which learned the weights to assign to each individual model, thus creating a weighted ensemble. This approach resulted in a 0.1631 RMSE on the leaderboard and a RMSE of approximately 0.06 on the validation data. This means that the stack resulted in obvious overfitting.

We further experimented with the same approach using a decision tree in the last layer instead of a Linear regression, for learning the weights in the ensemble, however, this resulted in a more overfit model, with a RMSE of 0.1805 on the leaderboard.

Since this approach resulted in fast overfitting, we decided to abandon it.

The other method of ensembling the models that we attempted was a simple weighted Ensemble. Using a combination of the Linear Regression models, the XGBoost models, the Random Forest models, and the feed-forward neural network models, we attempted to manually adjust the weights that these models had on the final outcome. The best Ensemble was found to be an equal weights Ensemble between the highest-scoring Random Forest model, and highest scoring XGBoost model, which had a validation RMSE of 0.1318, and a test RMSE of 0.1586.

We then tried another ensemble, which used the underfitted feed-forward neural network with a weight of 0.2, the highest-scoring XGBoost model with a weight of 0.4, and the highest-scoring Random Forest model with a weight of 0.4, all trained on the features chosen with the Boruta search method. We expected that the feed-forward neural network would generally make mistakes in the opposite direction of the Random Forest and XGBoost models, and therefore contribute to the reduction of the average mistake. The weight of the feed-forward neural network model is low, however, due to its larger average errors. This resulted in an Ensemble with a validation RMSE of 0.1856, and a test RMSE of 0.1567, our best score in this competition.

F. Model evaluation

In this subsection, we present the results of the individual models which were optimized and chosen for creating ensembles in the final stage of experimentation. Table I shows the

models, their training configurations, the features they used, and their validation and leaderboard RMSE scores.

The final 3 models which were chosen for the competition include:

- Ensemble using the best performing feed-forward neural network, Random Forest, and XGBoost models
- Ensemble using only the best performing Random Forest and XGBoost models (excluding models which expect Gaussian distributions of the target variable)
- The best performing Random Forest which uses the features chosen with the Boruta search method to avoid over-smoothing or overfitting the ensemble methods

VII. DISCUSSION AND FUTURE WORK

The top-performing models were the XGBoost and Random Forest models, strongly outperforming the Linear Regression models and slightly outperforming the feed-forward neural network models. This was expected, due to the multi-modal nature of the target variable, which is hard to estimate using models that expect a Gaussian distribution of the target.

Moreover, the ensembles of diverse models performed the best out of all the predictive options. They used weights that were calculated using the inverse of the RMSE scores of the cross-validation of the models they were composed of. The singular Linear Regression models were not used in the Ensembles due to their massive underperformance compared to the other three model types. However, they were used with the bagging regressors, but this approach also underperformed compared to the non-linear approaches.

Hyper-parameter optimization on all models was performed using the grid search algorithm, with 5-fold cross-validation, and RMSE as a metric to evaluate performance. Grid search was used because it is one of the most thorough hyper-parameter tuning algorithms. Given more time, we would have expanded the search space of the grid search of all models.

According to the Boruta search for feature importance, the *eucledian_distance*, *direction_id*, and *km_total* columns were considered the most important for determining the *expenses* value, with starting and ending locations of low-frequency destinations being some of the least important features. This further implies that the distance of the route is the most important deciding factor in the final expenses of forwarding contracts.

The main challenge in working with this dataset was the limited information we had on the meaning of some of the given features. With better information on the features, the data engineering process, as well as the model building process, would have been more specific and exhaustive.

While experimenting with the aforementioned validation procedures in section III, we noticed that some additional features could be extracted from the *id_payer* column, considering that it, in its original form, is not applicable as a feature. Such derived features could be:

- *num_previous_contracts* - the number of previous contracts (before this contract date) for the same client (*id_payer*)

- *average_cost_previous_contracts* - the average cost of previous contracts (before this contract date) for the same client (*id_payer*)
- *average_duration_previous_contracts* - the average duration of previous contracts (before this contract date) for the same client (*id_payer*)
- *average_length_previous_contracts* - the average length of previous contracts (before this contract date) for the same client (*id_payer*)
- *ratio_length_previous_contracts* - the ratio of current length divided by the average of previous contracts (before this contract date) for the same client
- *cost_most_similar_contract* - the cost of the previous contract with the most similar length, adjusted by the difference in exchange ratios

Considering that computation of such features should be properly handled and should be closely integrated with the training-validation split process, we did not utilize them. Despite that, we believe that there is merit in further experimenting with them.

Although it was considered, fuel prices were ultimately not used in the prediction of the target variable. This was due to the uncertainty of the availability of current fuel prices, making using them a potential data leak.

Finally, the usage of external public datasets could have vastly improved the predictions of all models. Unfortunately, due to time restrictions, we were unable to properly search for, test, and use any relevant public dataset.

VIII. CONCLUSION

The original goal of the challenge was to use preprocessing methodologies, Machine Learning algorithms and feature selection methods, in order to most accurately predict the costs related to the execution of forwarding contracts in a transporting company. Using feature engineering techniques, as well as a weighted diversified ensemble of XGBoost, Random Forest, and deep learning models (a feed-forward neural network), we were able to predict the expenses of the forwarding contracts with a RMSE of 0.1573.

In this paper, all missing data was imputed using mean filling and median filling. However, in the future, more sophisticated methods for data imputation can be utilized, such as Multiple Imputation by Chained Equations [40] or Regression Imputation [41].

In a broader context, we can conclude that in real-life business problems, domain knowledge and information are essential. With manual feature extraction that reflects the domain knowledge, valuable features could be created that improve the model performance. Likewise, without the domain knowledge, the model validation from a practicality and explainability perspective could be limited.

ACKNOWLEDGEMENT

This work was partially financed by the Faculty of Computer Science and Engineering at the Ss.Cyril and Methodius University, Skopje, Macedonia.

REFERENCES

- [1] E. Zdravevski, P. Lameski, C. Apanowicz, D. Slezak, From big data to business analytics: The case study of churn prediction, *Applied Soft Computing* 90 (2020) 106164. doi:https://doi.org/10.1016/j.asoc.2020.106164.
- [2] J. Bi, C. Zhang, An empirical comparison on state-of-the-art multi-class imbalance learning algorithms and a new diversified ensemble learning scheme, *Knowledge-Based Systems* 158 (2018) 81–93. doi:https://doi.org/10.1016/j.knsys.2018.05.037.
- [3] M. Grzegorowski, E. Zdravevski, A. Janusz, P. Lameski, C. Apanowicz, D. Slezak, Cost optimization for big data workloads based on dynamic scheduling and cluster-size tuning, *Big Data Research* 25 (2021) 100203. doi:https://doi.org/10.1016/j.bdr.2021.100203.
- [4] E. Zdravevski, P. Lameski, A. Kulakov, S. Filiposka, D. Trajanov, B. Jakimovski, Parallel computation of information gain using hadoop and mapreduce, in: 2015 Federated Conference on Computer Science and Information Systems (FedCSIS), IEEE, 2015, pp. 181–192.
- [5] A. Janusz, A. Jamiołkowski, M. Okulewicz, Predicting the costs of forwarding contracts: Analysis of data mining competition results, in: *Proceedings of the 17th Conference on Computer Science and Intelligence Systems, FedCSIS 2022, Sofia, Bulgaria, September 4-7, 2022*, IEEE, 2022.
- [6] A. Janusz, G. Hao, D. Kaluza, T. Li, R. Wojciechowski, D. Slezak, Predicting escalations in customer support: Analysis of data mining challenge results, in: 2020 IEEE International Conference on Big Data (Big Data), IEEE, 2020, pp. 5519–5526.
- [7] A. Janusz, M. Przyborowski, P. Biczuk, D. Slezak, Network device workload prediction: A data mining challenge at knowledge pit, in: 2020 15th Conference on Computer Science and Information Systems (FedCSIS), IEEE, 2020, pp. 77–80.
- [8] A. Janusz, D. Kaluza, A. Chkadzyska-Krasowska, B. Konarski, J. Holland, D. Slezak, Ieee bigdata 2019 cup: suspicious network event recognition, in: 2019 IEEE International Conference on Big Data (Big Data), IEEE, 2019, pp. 5881–5887.
- [9] M. Matraszek, A. Janusz, M. Swiechowski, D. Slezak, Predicting victories in video games-ieee bigdata 2021 cup report, in: 2021 IEEE International Conference on Big Data (Big Data), IEEE, 2021, pp. 5664–5671.
- [10] P. Refaeilzadeh, L. Tang, H. Liu, Cross-validation., *Encyclopedia of database systems* 5 (2009) 532–538.
- [11] G. Behera, N. Nain, Grid search optimization (gso) based future sales prediction for big mart, in: 2019 15th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), IEEE, 2019, pp. 172–178.
- [12] S. Punia, K. Nikolopoulos, S. P. Singh, J. K. Madaan, K. Litsiou, Deep learning with long short-term memory networks and random forests for demand forecasting in multi-channel retail, *International journal of production research* 58 (16) (2020) 4964–4979.
- [13] A. Dutta, A. Dureja, S. Abrol, A. Dureja, et al., Prediction of ticket prices for public transport using linear regression and random forest regression methods: A practical approach using machine learning, in: *International Conference on Recent Developments in Science, Engineering and Technology*, Springer, 2019, pp. 140–150.
- [14] S.-J. Joo, H. Min, C. Smith, Benchmarking freight rates and procuring cost-attractive transportation services, *The International Journal of Logistics Management* (2017).
- [15] Z. Yang, E. E. Mehmed, Artificial neural networks in freight rate forecasting, *Maritime Economics & Logistics* 21 (3) (2019) 390–414.
- [16] A. Ubaid, F. Hussain, J. Charles, Modeling shipment spot pricing in the australian container shipping industry: case of asia-oceania trade lane, *Knowledge-based systems* 210 (2020) 106483.
- [17] S. Nataraj, C. Alvarez, L. Sada, A. Juan, J. Panadero, C. Bayliss, Applying statistical learning methods for forecasting prices and enhancing the probability of success in logistics tenders, *Transportation Research Procedia* 47 (2020) 529–536.
- [18] Z. Men, E. Yee, F.-S. Lien, D. Wen, Y. Chen, Short-term wind speed and power forecasting using an ensemble of mixture density neural networks, *Renewable Energy* 87 (2016) 203–211.
- [19] X. Ma, J. Sha, D. Wang, Y. Yu, Q. Yang, X. Niu, Study on a prediction of p2p network loan default based on the machine learning lightgbm and xgboost algorithms according to different high dimensional data cleaning, *Electronic Commerce Research and Applications* 31 (2018) 24–39.
- [20] C.-Y. Wang, M.-Y. Lin, Prediction of accrual expenses in balance sheet using decision trees and linear regression, in: 2016 Conference on Technologies and Applications of Artificial Intelligence (TAAI), IEEE, 2016, pp. 73–77.
- [21] T. Chen, T. He, M. Benesty, V. Khotilovich, Y. Tang, H. Cho, K. Chen, et al., Xgboost: extreme gradient boosting, *R package version 0.4-2* 1 (4) (2015) 1–4.
- [22] S. J. Rigatti, Random forest, *Journal of Insurance Medicine* 47 (1) (2017) 31–39.
- [23] E. Zdravevski, P. Lameski, R. Mingov, A. Kulakov, D. Gjorgjevikj, Robust histogram-based feature engineering of time series data, in: 2015 Federated Conference on Computer Science and Information Systems (FedCSIS), 2015, pp. 381–388. doi:10.15439/2015F420.
- [24] E. Zdravevski, P. Lameski, A. Kulakov, S. Kalajdziski, Transformation of nominal features into numeric in supervised multi-class problems based on the weight of evidence parameter, in: 2015 Federated Conference on Computer Science and Information Systems (FedCSIS), 2015, pp. 169–179. doi:10.15439/2015F90.
- [25] P.-E. Danielsson, Euclidean distance mapping, *Computer Graphics and image processing* 14 (3) (1980) 227–248.
- [26] N. R. Chopde, M. Nichat, Landmark based shortest path detection by using a* and haversine formula, *International Journal of Innovative Research in Computer and Communication Engineering* 1 (2) (2013) 298–302.
- [27] I. J. Myung, Tutorial on maximum likelihood estimation, *Journal of mathematical Psychology* 47 (1) (2003) 90–100.
- [28] J. Benesty, J. Chen, Y. Huang, I. Cohen, Pearson correlation coefficient, in: *Noise reduction in speech processing*, Springer, 2009, pp. 1–4.
- [29] G. Chandrashekar, F. Sahin, A survey on feature selection methods, *Computers & Electrical Engineering* 40 (1) (2014) 16–28.
- [30] M. B. Kursu, W. R. Rudnicki, Feature selection with the boruta package, *Journal of statistical software* 36 (2010) 1–13.
- [31] P. Crewson, *Applied statistics handbook*, AcaStat Software 1 (2006) 103–123.
- [32] E. Winter, The shapley value, *Handbook of game theory with economic applications* 3 (2002) 2025–2054.
- [33] T. Jayalakhmi, A. Santhakumaran, Statistical normalization and back propagation for classification, *International Journal of Computer Theory and Engineering* 3 (1) (2011) 1793–8201.
- [34] G. D. Hutcheson, Ordinary least-squares regression, L. Moutinho and GD Hutcheson, *The SAGE dictionary of quantitative management research* (2011) 224–228.
- [35] H. A. Chipman, E. I. George, R. E. McCulloch, Bayesian cart model search, *Journal of the American Statistical Association* 93 (443) (1998) 935–948.
- [36] A. F. Agarap, Deep learning using rectified linear units (relu), arXiv preprint arXiv:1803.08375 (2018).
- [37] I. Nusrat, S.-B. Jang, A comparison of regularization techniques in deep neural networks, *Symmetry* 10 (11) (2018) 648.
- [38] H. Zheng, Z. Yang, W. Liu, J. Liang, Y. Li, Improving deep neural networks using softplus units, in: 2015 International joint conference on neural networks (IJCNN), IEEE, 2015, pp. 1–4.
- [39] T. G. Dietterich, Ensemble methods in machine learning, in: *International workshop on multiple classifier systems*, Springer, 2000, pp. 1–15.
- [40] I. R. White, P. Royston, A. M. Wood, Multiple imputation using chained equations: issues and guidance for practice, *Statistics in medicine* 30 (4) (2011) 377–399.
- [41] Z. Zhang, Missing data imputation: focusing on single imputation, *Annals of translational medicine* 4 (1) (2016).

APPENDIX A

APPENDIX 1: CONFIGURATION FOR THE INDIVIDUAL MODELS

TABLE I
RESULTS AND CONFIGURATION FOR THE INDIVIDUAL MODELS.

Id	Algorithm	Params	Features / preprocessing	Total Features	Train config	RMSE val	RMSE leaderboard
1	LinearRegression	/	[euro_exchange_rate, first_load_lat, last_unload_lat, first_load_lon, last_unload_lon, route_start_lat, route_end_lat, route_start_lon, route_end_lon, km_empty, km_nonempty, train_km, route_duration, km_total, max_weight, route_start_numeric, time_taken_minutes] - STANDARDIZED [ferry_intervals, train_intervals, id_service_type] - MIN_MAX_SCALED select from above preprocessed features	43	validation split based on id_payer	0.6703012807	0.4598
2	LinearRegression	/	[euro_exchange_rate, first_load_lat, first_load_lon, route_end_lat, last_unload_lon, km_empty, km_total, train_km, ferry_duration, max_weight, route_start_numeric, time_taken_minutes] add km_total_sq, max_weight_sq and time_taken_minutes_sq to above features	12	validation split based on id_payer	0.6942584157	0.5027
3	LinearRegression	/	add km_total_cube, max_weight_cube and time_taken_minutes_cube to above features	15	validation split based on id_payer	0.5713369216	0.4309
4	LinearRegression	/	Predicted using XGBRegressor with hyper-parameter tuning on alpha, booster, eta, lambda and max_depth.	18	validation split based on id_payer	0.5092784726	1.0127
6	XGBoost	/	Extracted aggregate features from routes table	43	validation split based on id_payer	0.34	0.1735
7	XGBoost	max_depth = 10, other default	Feature selection with Boruta shap	73	5-fold cv RMSE averaged	0.18	0.1649
8	XGBoost	Dense(64, activation=relu) Dropout(0.2)		48	BATCH_SIZE = 32 EPOCHS = 500	0.15	0.1622
9	Feed Forward Network	BatchNormalization() Dropout(0.2) Dense(128, activation=relu) Dropout(0.2) BatchNormalization() Dense(1, activation=softplus)	Min/Max scaling all features except binary	48	LEARNING_RATE = 10e-3 OPTIMIZER = Adam EARLY_STOPPING on validation RMSE patience 3 VALIDATION_SPLIT = 0.2	0.2911	0.2895
10	Feed Forward Network	Dense(128, activation=relu) Dropout(0.2) BatchNormalization() Dense(64, activation=relu) Dropout(0.2) BatchNormalization() Dense(64, activation=relu) Dropout(0.2) BatchNormalization() Dense(1, activation=softplus)	Min/Max scaling all features except binary	48	BATCH_SIZE = 16 EPOCHS = 50 LEARNING_RATE = 10e-4 OPTIMIZER = Adam EARLY_STOPPING on validation RMSE patience 3 VALIDATION_SPLIT = 0.2	0.183	0.193
11	Feed Forward Network	Dense(128, activation=relu) Dense(64, activation=relu) Dense(64, activation=relu) Dense(1, activation=linear)	Min/Max scaling all features except binary	48	BATCH_SIZE = 16 EPOCHS = 50 LEARNING_RATE = 10e-4 OPTIMIZER = Adam EARLY_STOPPING on validation RMSE patience 3 VALIDATION_SPLIT = 0.2	0.1683	0.1775
12	Random Forest	max_depth = 10, min_samples_leaf = 5, max_features = 40	All features used	40	5-fold cv RMSE averaged	0.1662	0.1726
13	Random Forest	max_depth = 10, min_samples_leaf = 5, max_features = 40, n_samples = 200, min_samples_leaf = 1	Feature selection with Boruta shap	40	5-fold cv RMSE averaged	0.1594	0.1625

Key Factors to Consider when Predicting the Costs of Forwarding Contracts

Quang Hieu Vu
 ZaloPay, VNG Corporation, Vietnam
 hieuvq@vng.com.vn

Ling Cen, Dymitr Ruta, Ming Liu
 EBTIC, Khalifa University, UAE
 {cen.ling,dymitr.ruta,liu.ming}@ku.ac.ae

Abstract—Predicting the cost of forwarding contracts is a typical problem that logistics companies need to solve in order to optimize their business for a better profit. This is the challenge defined in the FedCSIS 2022 Competition where a five-year history of contract data and their delivery routes from a large Polish logistics company are provided to train a Machine Learning model. In addition to the contract data, historical wholesale fuel prices and euro exchange rates at the contract time are also provided. To address this challenge, we first designed a basic solution where we focused on feature engineering to find good impact features for the model. After that, the same set of features were used to train two different models: one using XGBoost and the other using LightGBM. The average predictions of the two boosting models were then used as the predictions for the next post-processing step. Finally, in the post-processing step, we designed and trained a simple linear regression model to capture the average monthly changes of the contract cost, given the changes of the fuel prices and euro exchange rates. These captured changes were used to post-process (adjust) the predictions in the previous step to address the issue that tree-based models could not predict the value that they did not see before. While the basic solution with careful feature selection gave us a place in the top-5, our post-processing strategy in the last step helped us win the 3rd prize in the competition.

Index Terms—Logistics, Forwarding Contract Cost Prediction, Gradient Boosting Trees, XGBoost, LightGBM, Linear Regression, Feature Engineering, Post-Processing.

I. INTRODUCTION

MANY logistics companies are on the road to digital transformation and employ AI/Machine Learning technologies to support and optimize their daily business. One of the key challenges that these companies are facing is to predict to cost of a delivery/forwarding contract in which an accurate prediction result can help the logistics companies in many aspects of which three typical ones are listed below:

- To maximize revenue by selecting profitable contracts.
- To identify issues and optimize operations to reduce costs.
- To get better planning in contract execution.

The challenge of predicting the cost of forward contract is exactly the objective of the FedCSIS 2022 [1] competition ¹, which is in cooperation with PTI and QED Software and sponsored by Control System Software ² – a software company that has been delivering solutions for the Transportation, Spedition, and Logistics industry for 20 years. In this competition, a

five-year history of contract data and their delivery routes together with wholesale fuel prices and euro exchange rates at the contract time are provided. Specifically, two types of data sources are provided. While the first data source contains basic information about the contracts, the second one describes the main sections of the planned routes associated with each contract. Given the data, when we performed exploration data analysis (EDA), we could see that the amount of data is not big, there are not much information in the contract to analyse, and the number of provided features in the contract data is small. As a result, we believed that the most important factor to design and train a good prediction model is in the feature engineering process where (1) We need to generate extra features to be used, in addition to the existing provided features and (2) useful features should be carefully selected for the model. With respect to this point, our first contribution in this paper is in this feature engineering process. Once we got the selected features, we then simply trained two boosting tree models: XGBoost and LightGBM and obtained the average predictions from the two models as the forecast.

Our second contribution in this paper is a simple and effective approach to post-process the predictions to capture the trends in contract costs. Basically, it is well-known that using tree-based models in the presence of trends over time can lead to inaccurate results. The reason is in the way tree-based models make predictions. For example, Decision Trees make regression predictions by seeing which “leaf” the data point belongs to and assigning the average of the target variable from the training set to that point. In this case, they fail to accurately predict values they have not already seen, and values from a significantly different population (perhaps after some trend in time) will cause the model to make inaccurate predictions. Random Forests and Gradient Boosting Tree algorithms suffer the same problem because their results are averages results of Decision Trees. In this competition, when we performed EDA, we could see the fuel prices have steadily increased in the past couple of years. Euro exchange rates have also increased during the time of the test set. As a result, there should be an increase trend in the average cost of the contracts in the test set, which may not be well captured by the models in the previous step. Therefore, we designed and trained a simple linear regression model to capture the trend in the average cost of forwarding contracts and use this model’s result to post-process the predictions from tree-based models. This post-

¹<https://knowledgepit.ml/fedcsis-2022-challenge/>

²<https://controlsystem.com.pl/>

processing step did help to uplift our prediction model and brought us to the 3rd position in the final ranking list.

To summarize, our contributions are twofold:

- We present our feature engineering process in which we first share how to generate extra features using route information and then select good features having impacts to the forecast model from a set of candidate features.
- We introduce a simple linear regression model used in the post-processing step to overcome an issue of tree-based models in capturing trends in contract costs.

For the rest of the paper, we organize the content as follows. The background and related work are introduced in Section II. An overview of the competition challenge is described in Section III. The details of the proposed method are presented in two sections, where Section IV is dedicated for feature engineering and feature selection while Section V shows details of the model design, focusing on the linear regression model used in the post-processing step. Finally, conclusions are given in Section VI.

II. RELATED WORK

The costs of forwarding contracts can be affected by many factors, which, besides contract nature and transportation arrangement, fuel prices, currency exchange trends are decisive too. The actual transportation cost is also constrained by some factors such like behavior of drivers, weather, traffic, market demand, etc., accurately forecasting the cost is, thus, challenging, but critical in the business providing logistics service or involving supply chains [5].

Artificial intelligence techniques empowered solutions have been investigated to solve transportation problems in both industry and academy. A logit neural network (NN) based mode selection model was developed to address border transportation in [11]. Freight demand was predicted for inter-modal terminals by NNs in [12]. Traffic volume in non-urban highways under heterogeneous conditions and vehicle counts were predicted by developing a NNs based model in [13].

There is, however, quite little work published in the literature on cost prediction of forwarding contracts. The work in [14] developed AI based models to predict the long-term cost of the logistics service, and attempted to construct a risk-aware interval for the prices to be offered in the bid, aiming to boost competitiveness in the application for tenders, and in addition, historical data was used to develop statistical learning models for predicting the success likelihood of a tender based on the actual data and predicted service prices achieved from previous stage. In [5], a trapezoidal neutrosophic fuzzy analytical hierarchy process (TNF-AHP) was proposed to determine the most significant criteria that were used to predict transportation cost by an artificial neural network (ANN) model, which, claimed by the authors, can be also used in supply chain management and inventory control management.

Boosting is a popular technique used in machine learning to reduce errors in predictions from which to improve the accuracy of machine learning models. The basic idea of boosting approach is to combine a set of weak models into a strong

and robust model, which is able to reduce the prediction bias [6], [7]. Gradient boosting (GB) is an extension of boosting, in which the process of additive generation of weak models is based on a gradient descent algorithm over an objective function [8]. Gradient boosting decision tree (GBDT) is an ensemble model of decision trees, used as weak learners in the gradient boosting ensemble model. In each iteration of its training process, a new decision tree is added to the pool, which tries to increasingly learn on mistakes of decision trees already in the pool by fitting the negative gradients (or residual errors) [9], [10].

III. COMPETITION DESCRIPTION

The challenge of the FedCSIS 2022 [1] is to predict the cost of forward contracts, using five-year history of contract data and their delivery routes together with wholesale fuel prices and euro exchange rates at the contract time. The data is provided in three different files as follows:

- Main contract data: contain general information of forward contracts that are ready to be used as features to train a model. The data is stored in two .csv files: “css_main_training.csv” and “css_main_test.csv”.
- Contract route data: provide detailed information about routes of forward contracts, which can be used to generate extra features (note that some basic route features are already created in the main contract data). The data is stored in two .csv files: “css_routes_training.csv” and “css_routes_test.csv”.
- Wholesales fuel prices: store average fuel prices at monthly level throughout the time of forward contracts in a single .csv file, “fuel_prices.csv”.

The accuracy of the prediction model is measured by the Root Mean Square Error (RMSE) metric. Note that the purpose of using RMSE is to give higher penalty for large errors compared to smaller ones because the errors are squared before they are averaged.

IV. FEATURE ENGINEERING

The process of feature engineering, which typically includes feature generation at first, followed by feature selection using Greedy Forward Search, Greedy Backward Search, and finally SHAP analysis [15] for feature importance, is clearly presented in this paper [16]. Thus, in this section, we simply discuss which sources we used to generate our features as well as the list of our selected features.

In general, the features that were used to train our models to predict the cost of forward contracts were generated in the two basic steps. In the first step, we executed “feature generation” during which extra features are generated from the first two data sources: “main training data” and “contract route data”. After that, we performed “feature selection” to determine the usefulness of generated features based on their impact on the performance of our models and finally chose which are the features that should be included in training our final models. In general, our features could be classified into “basic features” which are available in the “main contract

data” and “extra features” which are generated in the feature generation process.

A. Basic features

We use all available features in the “main training data” but one and they are all good features in the top-20 important features returned by our model. The only exception is from the “euro_exchange_rate”, which we thought to be an important feature, but the model told us otherwise. We tried a number of ways to utilize the “euro_exchange_rate” such as using it as a single feature, identifying trend up or down in the exchange rates, or converting it into a mean-encoding feature. But, none of the above approaches was succeeded. As a result, we did not use the “euro_exchange_rate” in the main model. However, it turned out that this feature was still useful in the linear regression model that we employed later in our post-processing step presented later in Section V.

B. Extra features

Extra features are mainly generated from “css_routes” files. These features help to identify special properties of the routes from which giving the model better forecast accuracy. They include the following sets of features:

- The route statistics features: include the number of route segments, the number of route segments with and without cargo, and the number of route segments where starting and ending points are in the same country (or in different countries). These features give us extra information on how big or complexity the forward contract is, in addition to the existing “km distance” feature.
- The gap feature between “km distance” and “Haversine distance”: this feature provides us how easy or hard a contract forward can be executed. Picture that if the “Haversine distance” is short while the “km distance” is long, the implication is that it is not easy or straightforward to move directly from the start to the end of the contract as some detour may be required in the trip.
- The statistics of vehicles and trailers used in the trip: include the average of axle counts from all vehicles and trailers, the average of kerb weight from all vehicles and trailers, the average of vehicle engine capacity and the average of trailer payload. These features tell us how heavy the cargo is in the trip, in addition to the “max weight” feature in the main contract data.
- The features about distance between the route start and the point where the cargo is loaded first as well as distance between the route end and the point where the cargo is last unloaded. These features tell us the percent utilization of vehicles in the contract.

In addition to route features, as we have a text describing the temperature requirement in the main contract data, we generated the following temperature requirement features, which are good features for the model because the forward contract cost should be higher if there are special requirements for the temperature such as frozen or automatic. Specifically, these features include:

- The range of temperature (low and high) if they are mentioned in the text field. Otherwise, they are left with NULL values. In some cases, where a fixed temperature is required, we set equal values for both the low temperature and high temperature.
- One-hot encoding features for the top-5 important words detected from the text such as frozen, continuous, or automatic. These words are created simply by getting the top-5 words returned from “CountVectorizer” after removing stop words.

Finally, as we have the time when the trip starts and ends in the main contract data, we generate these following extra time related features:

- The weekdays where the trip starts and ends as the cost could be different if we start or end the trip at different days of the week. For example, the trip starts or ends during the weekend may lead to a higher contract cost.
- Similar to the above case, the hour of day where the trip starts and ends also have impacts on the contract cost. A night time start or end is expected to have a higher cost compared to other time of the day.
- Finally, the day of month where the trip starts and ends and the duration (in terms of hours and days) of the trip also contribute to changes in the contract cost.

C. Model design and implementation

Given the list of good features obtained in the above feature engineering step, we simply designed and trained two boosting trees: XGBoost and LightGBM, using the same set of features and the final prediction is the average predictions returned from the two models. As you can expect, the combination of the two models using the same set of features does not help to make much improvement in the model accuracy. Instead, we employ this strategy simply to get a stable prediction result as our main concern is always the overfitting issue given the small amount of data used for the public leaderboard, which we experienced in the past two competitions of 2020 [2], [3] and was presented in the paper [4].

V. FORECAST POST-PROCESSING WITH LINEAR REGRESSION

As discussed in Section IV, we could not utilize the euro exchange rates in the main model. Similarly, when we tried to add wholesale fuel prices to the main model, it does not have a positive impact. However, when we looked at the changes in both euro exchange rates and fuel prices, we could see that during the period of testing, there was a significant increase in both euro exchange rates and fuel prices. As discussed in Section I, since using tree-based models in the presence of trends over time can lead to inaccurate results, this section introduces our solution to address this issue. Specifically, we will present a design and implementation of a simple linear regression model to predict monthly changes in the average contract costs in Section V-A and then use the predicted result to adjust the predictions returned by the main model in Section V-B.

A. Forecasting trend in contract costs

To detect trend in contract costs, we built a simple model to detect the correlation between changes in euro exchange rates and fuel prices and changes in the average contract cost at the monthly level. Note that here we assume that the types of executing contracts each month follow a similar distribution. As the impact of euro exchange rates and fuel prices could be different given different trip distance (e.g., short distance trip may be more or less sensitive to the euro exchange rates and fuel price changes compared to long distance trip) and whether train routes or ferry routes are involved in the trip, we need to consider this factor into the model. In the end, we trained a linear regression model using the following features aggregated at the monthly level:

- The minimum, mean, and maximum of euro exchange rates as well as fuel prices.
- The distance group which we defined based on the total km of the trip. In our solution, we split contracts equally into 4 groups having total km greater than 820, between 430 and 820, between 230 and 430, and less than 230.
- The last features are indicators of whether the trip has train routes or ferry routes.

The training target of the model is the average contract costs each month obtained from the training data set. Once the model is trained, we use it to predict the average contract costs for months in the test data set.

B. Post-processing predictions

Given the predicted average contract cost of a month returned from the above linear regression model (let's call it A), we compare it against the average predicted contract costs returned from our main model trained with boosting trees presented in Section IV (let's call it B), three cases may happen:

- If A is equal to B , our two prediction models are aligned. It is good and we should not do anything for post-processing.
- In cases A is less than B , there could be a down trend in the average contract costs that fail to be captured by our tree-based models, and hence the models generate over-forecast. In this case, we first compute $\delta = B - A$ and then subtract δ from all predictions returned from the main model.
- In cases A is greater than B , it is opposite as there could be an up trend that the tree-based models fail to capture, and hence we need to add in a $\delta = A - B$ for all predictions made by the main model.

It is interesting to note that in our case, the linear regression model always returned a higher prediction for the average contract costs. It means that there should be an expected increase in the contract costs, given the hike of the euro exchange rates and the fuel prices during the period of time used for testing. In our solution, by applying this post-processing step, we could see a score improvement of 0.006 in the Public Leader Board,

which is a good improvement, given that even good features could only help to improve score of 0.002 to 0.003.

VI. CONCLUSIONS

In this paper, we presented our solution to win the 3rd prize of the FedCSIS 2022 Challenge. There are two key factors leading to the effectiveness of our solution. The first one is a process of feature engineering to generate extra useful features and then carefully select features for the model. The second one is a solution for post-processing the predictions in order to capture changing trends in the forward contract costs, which otherwise are failed to get in the main tree-based models. They were both discussed in the paper.

REFERENCES

- [1] A. Janusz, A. Jamiolkowski, M. Okulewicz, "Predicting the Costs of Forwarding Contracts: Analysis of Data Mining Competition Results", *Proceedings of the 17th Conference on Computer Science and Intelligent Systems (FedCSIS)*, 2022.
- [2] A. Janusz, M. Przyborowski, P. Biczuk, D. Ślęzak, "Network Device Workload Prediction: A Data Mining Challenge at Knowledge Pit", *Proceedings of the 15th Conference on Computer Science and Information Systems (FedCSIS)*, 2020, doi: 10.15439/2020F159.
- [3] A. Janusz, G. Hao, D. Kaluza, T. Li, R. Wojciechowski, D. Ślęzak, "Predicting Escalations in Customer Support: Analysis of Data Mining Challenge Results", *IEEE International Conference on Big Data*, 2020, doi: 10.1109/BigData50022.2020.9378024.
- [4] D. Ruta, L. Cen, Q. H. Vu, "Deep Bi-Directional LSTM Networks for Device Workload Forecasting", *Proceedings of the 15th Conference on Computer Science and Information Systems (FedCSIS)*, 2020, doi: 10.15439/2020F213.
- [5] A. Singh, A. Das, U. K. Bera, G. M. Lee, "Prediction of Transportation Costs Using Trapezoidal Neutrosophic Fuzzy Analytic Hierarchy Process and Artificial Neural Networks", *IEEE Access*, vol. 9, 2021, doi: 10.1109/ACCESS.2021.3098657.
- [6] L. Breiman, "Bagging predictors", *Machine Learning*, vol. 24, no. 2, pp. 123-140, 1996.
- [7] L. Mason, J. Baxter, P. Bartlett, and M. Frean, "Boosting Algorithms as Gradient Descent", in S.A. Solla, T.K. Leen, and K.R. Muller, editors, *Advances in Neural Information Processing Systems 12*, MIT Press.
- [8] L. Mason, J. Baxter, P.L. Bartlett, and M. Frean, "Boosting algorithms as gradient descent", *Proceedings of International Conference on Neural Information Processing Systems*, MIT Press, 1999.
- [9] J. H. Friedman, "Greedy function approximation: A gradient boosting machine", *Ann. Statist.*, vol. 29, no. 5, pp. 1189-1232, 2001.
- [10] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T. Liu, "LightGBM: A Highly Efficient Gradient Boosting Decision Tree", *Proc. Neural Information Processing Systems Conference (NIPS)*, 2017.
- [11] U. Gazder and N. T. Ratrouf, "A new logit-artificial neural network ensemble for mode choice modeling: A case study for border transport", *J. Adv. Transp.*, vol. 49, no. 8, pp. 855-866, 2015.
- [12] I. C. Bilegan, T. G. Crainic, and M. Gendreau, "Forecasting freight demand at intermodal terminals using neural networks—an integrated framework", *Eur. J. Oper. Res.*, vol. 13, no. 1, pp. 22-36, 2008.
- [13] K. Kumar, M. Parida, and V. K. Katiyar, "Short term traffic flow prediction for a non urban highway using artificial neural network", *Procedia - Social and Behavioral Sciences*, vol. 104, pp. 755-764, 2013.
- [14] S. Nataraj, C. Alvarez, L. Sadaa, A. Juana, J. Panadero, C. Bayliss, "Applying Statistical Learning Methods for Forecasting Prices and Enhancing the Probability of Success in Logistics Tenders", *Transportation Research Procedia (Elsevier)*, vol. 47, 2020.
- [15] S. Lundberg, S. Lee, "A Unified Approach to Interpreting Model Predictions", *Advances in Neural Information Processing Systems 30 (NIPS)*, 2017.
- [16] Q. H. Vu, D. Ruta, L. Cen, M. Liu, "A combination of general and specific models to predict victories in video games", *IEEE International Conference on Big Data (Big Data)*, 2021, doi: 10.1109/Big-Data52589.2021.9671285.

An Approach for Predicting the Costs of Forwarding Contracts using Gradient Boosting

Haitao Xiao^{1,2}, Yuling Liu^{1,2*}, Dan Du¹, Zhigang Lu^{1,2}

¹*Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China*

²*School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China*

{xiaohaitao, liuyuling, dudan, luzhigang}@iie.ac.cn

Abstract—Predicting the cost of forwarding contract is a severe challenge to road transport management system. The transportation cost of a forwarding contract often depends on many factors. It is hard for humans to evaluate the various factors in transportation and calculate the cost of forwarding contract. In this paper, we propose an approach to address such a problem by following the sequence of machine learning steps which consist of data analysis, feature engineering and model construction. First, we conduct a detailed analysis of the given data. Then, we generate effective features to characterize the cost of forwarding contract and eliminate redundant features. Finally, in the model construction phase, we propose a gradient boosting decision tree based method to train and predict the cost of forwarding contract. The proposed approach achieves RMSE scores of 0.1391 on the test set, which is the 2nd final score in the competition.

Index Terms—cost prediction, gradient boosting, model ensemble

I. INTRODUCTION

COST PREDICTION is widely used in various fields, such as transportation[1], cybersecurity[2], construction[3], and healthcare[4]. Cost prediction is generally a method of studying historical data and predicting future costs. Effective cost prediction can help businesses better control costs and adjust management strategies for the future in a timely manner, allowing them to gain a competitive advantage. Transportation cost prediction is one of the aspects of cost prediction. Anitha and Patil [1] predicted the transportation costs using a regression algorithm, which assisted the retail sector predict the cost incurred for logistics. In the freight company, predicting the costs of freight forwarding contracts by using historical data can help freight companies better understand the causes of costs and select profitable contracts. Thus, from both academia and industry, predicting transportation costs has drawn a lot of attention. Accurate transportation cost prediction is still a challenging problem.

Based on the same background, the FedCSIS 2022 Challenge: Predicting the Costs of Forwarding Contracts[5] released the task to develop a predictive model that assesses the actual costs of individual orders as accurately as possible. In this challenge, six years of history of contract data and planned routes are provided. The key to the problem is to find the factors related to the cost.

* Corresponding author

In this paper, we propose an approach for predicting the cost of forwarding contracts using the gradient boosting method. First, we analyse the data given in this challenge and have a clear understanding of the meaning of each feature. The data analysis step also provides guidance for the following feature engineering. Then, in feature engineering, we generate the features that can effectively characterise the costs of forwarding contracts from contract data, planned routes, and historical wholesale fuel prices. We also remove redundant features after feature generation. The redundant feature elimination can reduce training time and the impact of noise on the training model. Finally, in the model construction phase, we propose a gradient boosting based method to train and predict the costs related to the execution of forwarding contracts. We introduce the model stacking mechanism as an ensemble method to enhance generalisation performance, which is a frequently used strategy in machine learning competitions. The experiments and competition results have both shown the effectiveness of the proposed approach.

In summary, this paper makes the following contributions:

- We analyse the given data and provide guidance for the following feature engineering. The guidance helps to generate effective features in feature engineering.
- We generate effective features from contract data, planned routes, and historical wholesale fuel prices for cost prediction. The generated features can improve the prediction performance significantly. And we also remove redundant features to reduce training time and the impact of noise on the training model.
- We propose an effective stacking approach using a gradient boosting based method to train and predict the costs of forwarding contracts, which achieves RMSE scores of 0.1402 on the preliminary testing subset and 0.1391 on the complete testing set. And we get the 2nd final score in the competition.

The remainder of this paper is structured as follows: Section II introduces the FedCSIS'22 challenge. Section III provides the analysis of the data and the details of our proposed approach. Section IV shows the results of the experiments. Finally, conclusions are drawn in Section V.

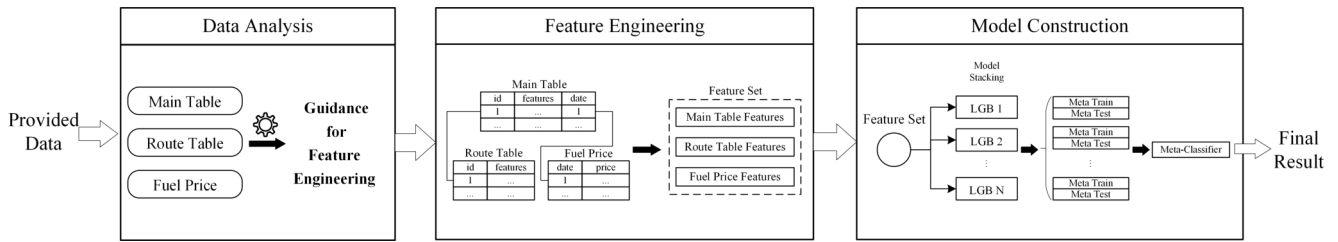


Fig. 1. Overview of proposed approach

II. FEDCSIS 2022 CHALLENGE

In this section, we will briefly introduce the FedCSIS 2022 Challenge, titled as Predicting the Costs of Forwarding Contracts.

The task in this challenge is to predict the costs related to the execution of forwarding contracts based on contract data and planned routes. Accurate cost prediction of forwarding contracts can support freight forwarders in selecting profitable contracts. The data provided in the challenge is collected by Control System Software, which is a software company that has been delivering solutions for the transportation, spedition, and logistics industry for 20 years[6].

The data set provided in this challenge contains a six-year history of orders appearing on the transport exchange. Details of the dataset are presented in the Section III. The aim of the competition is to develop a predictive model using the training set that assesses the actual costs of individual orders as accurately as possible.

The preliminary scores of submitted solutions are computed on a small subset of the testing data and published on the public leaderboard. The final evaluation is published using all of the test records after the competition ends.

III. METHODOLOGY

In this section, we describe in detail our proposed approach for predicting the cost of forwarding contracts. An overall framework of our proposed approach is provided in Figure 1. The approach includes data analysis, feature engineering, and model construction.

A. Data Analysis

TABLE I
BRIEF VIEW OF THE COMPETITION DATA

Data	Format	Size (train/test)
Main Table	csv	69.9 MB / 14.2 MB
Routes Table	csv	204 MB / 58.2 MB
Fuel Prices	csv	68.1 KB

There are two data tables and an additional set of data provided in this competition. A brief view of the data is shown in Table I. The main table contains basic information about the contracts, and the routes table describes the main sections of the planned routes associated with each contract. The main table and routes table are linked by “id_contract”. One contract usually consists of several route sections. The additional set

of data is fuel prices, which contains historical wholesale fuel prices for the period of training and test data.

The main table records the details of individual contracts. The main table contains unique contract ids and their related information, as well as the expenses of each contract. The associated information of a contract id includes 1) the general information of contract such as payer, currency, direction, contract type, service type, duty count, planned time. 2) the basic characteristics of the shipped goods, such as refrigeration, temperature, and maximum weight. 3) expected route information, such as longitude and latitude of loading and unloading positions, longitude and latitude of route start and end positions, kilometers to be covered according to the route plan, ferries and trains usage. As these data are important to predict the cost of forwarding contract, they form the core set of features used to train our model.

The route table records the main sections of each contract’s planned route. The route table has more detailed information about the planned route, which contains all route steps of each contract. The associated information of a route step includes 1) general information of route step such as the sequence number, step type, latitude and longitude of the end route step point, city, address, and estimated time. 2) information of ferry and train usage. 3) the vehicle and trailer status information. These data contain information about all of the route steps, which can help our model understand the specific route composition in a contract and assess the cost of a contract at the granularity of route step.

The fuel price table records historical wholesale fuel prices from 2016-01-01 to 2021-11-30, which covers the period of training and test data. This table describes three different types of fuel price information. The cost of transportation can be directly affected by the price of fuel. Thus, it is also an important component of freight forwarding contract costs.

Among these data tables, the main table provides the base information for the competition task. We can explore the properties for further feature engineering when we have a clear understanding of the meaning of the data and features. First, we analyse the expenses column, which is a continuous value to indicate the cost of forwarding contract. The min value of expenses is 2.879139, and the max value is 9.598065. It can be regarded as a regression task. Then, we find that the route table contains information about all route steps, and the fuel price table can indicate the fuel price at the time of the contract. The route table and fuel price table can provide more specific

information for predicting the cost of forwarding contract. Thus, it is necessary to extract more information related to the cost from the route table and fuel price table. This makes for more targeted feature engineering and more representative extracted features.

B. Feature Engineering

Following the data analysis phase described above, we generate three types of new features from the main table, route table, and fuel price table. We remove redundant features after all new features have been generated. In the following subsections, three types of new features and the process feature selection are comprehensively presented.

1) *Main Table Features*: Main table features are mainly generated from the main data table summarizing contract’s definition and can indicate the transportation costs. As the temperature feature is quite messy, we perform a simple data clean to correct the temperature in different formats. Then, we generate a series of new features to characterise the cost of contract. The generated features of main table can be divided into three parts: 1) basic features such as duration time, the haversine distance from route start to end location and load to unload location, the ratio of kilometers to be covered with empty trailer and so on. 2) cross features such as “direction×contract_type”, “first_load_country×last_unload_country, and so on. The cross features can characterize the links between different category features. 3) time features such as the year of route start time, the quarter of route start time, and so on. The time features can characterize the impact on costs at different times.

2) *Route Table Features*: Route table features are mainly generated from the route data table based on “id_contract”. Each contract has a different number of route sections. We aggregate the section attributes of each contract in different ways. For numerical attributes, we aggregate them by count, sum, and ratio operations. We also use statistical methods like mean, max, min, and median to summarise the numerical features of each contract. For category attributes, we aggregate them by counting their occurrences. Moreover, we count top 1000 cities and address to characterize the geographical situation of each contract.

3) *Fuel Price Features*: Fuel price features are mainly extracted from the fuel price data table. As it is generally to assess the cost before the contract starts, we directly merge the fuel price into each contract that matches the route start time. The merged fuel prices can represent current fuel price levels at the beginning of the contract.

4) *Feature Selection*: The feature set has over 2600 features after all the new features generated. We remove any features that are redundant or duplicate. A redundant feature is defined as the percentage of the value of a particular feature that is greater than 99.9%. Finally, we get 1092 features after simple feature selection.

C. Model Construction

Gradient boosting decision tree[7] is an model which uses decision tree as weak learner and improves model quality with

a boosting strategy[8]. The gradient boosting based method has been shown to achieve superior performance in various machine learning tasks, such as prediction[9] and ranking[10]. Due to its excellent performance and high accuracy, we choose the gradient boosting based method as our base model. There are multiple implementations of gradient boosting based method, like XGBoost[11], LightGBM[12], and CatBoost[13].

In model construction, we try various gradient boosting based methods to train the selected features and select the best method as the base model of our final solution. We finally choose LightGBM as our base model for its ability to handle the high dimensions of features and high efficiency. We also propose an ensemble approach, which introduces the model stacking mechanism, to improve the generalisation performance. The generalisation ability of an ensemble approach is usually much stronger than that of base learners[14].

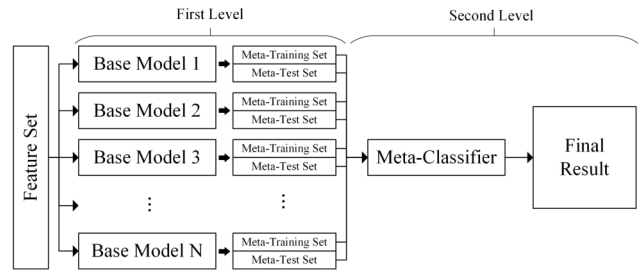


Fig. 2. The framework of proposed ensemble approach

The proposed ensemble approach can be separated into two levels, as shown in Figure 2. First, the training set is split into two parts, one part is used for first-level model training which is LightGBM, the other part, along with the test set, is used for prediction by the trained first-level model. Then, we use the predicted results from the first-level model to train the meta classifier. Typically, the meta classifier is a simple linear model. Therefore, we choose ridge regression as the meta classifier. The predicted result of the meta-classifier is our final result.

IV. EXPERIMENT AND RESULT

A. Experiment Setup

1) *Environment*: The operating environment is Ubuntu 21.10, memory at 128GB, an Intel Xeon Silver 4210 CPU @ 2.20GHz, with 40 physical processors.

2) *Toolkit*: For feature engineering implementation, we use Pandas 1.2.4, Numpy 1.20.1 to generate new features. We also use LightGBM 3.3.2 as the implementation of our base model.

3) *Evaluation Metric*: We take the root mean square error (RMSE) as the evaluation measurement, which is exactly the same as the evaluation criterion used for the competition.

B. Experiment Result

To ensure the scores between the local validation and the actual testing results remain within a certain range, it is important to have a good validation strategy. Cross-validation is a resampling method that uses different portions of the data to test and train a model on different iterations. In typical

cross-validation, the training and validation sets must cross over in successive rounds such that each data point has a chance of being validated against[15]. Since the labels are continuous values, we use the standard k -fold cross-validation as our local validation strategy. To determine the value of k , we tried different values of k and eventually found that the gap between local validation and online validation was smallest when $k = 3$. Thus, we set $k = 3$ and the gap between our local validation score and the public leaderboard score was less than 0.014.

In the experiment, we use different feature sets to evaluate each set's performance and the improvement of the model stacking. The experimental results are shown in Table II. In this table, "Baseline" represents the feature set contained in the original main data table, "FeatureSet₁" represents the merging feature set of the original main data table and generated main table features, "FeatureSet₂" represents the merging feature set of FeatureSet₁ and generated route table features, "FeatureSet₃" represents the merging feature set of FeatureSet₂ and generated fuel price features, and "Stacked" represents the model stacking based on the FeatureSet₃.

TABLE II
THE EXPERIMENTAL RESULTS OF DIFFERENT FEATURE SETS

Feature Set	RMSE Score(Local Validation)
Baseline	0.1463
FeatureSet ₁	0.1398
FeatureSet ₂	0.1276
FeatureSet ₃	0.1275
Stacked	0.1267

From Table II, it can be derived that: 1) From the result of training each feature set, the RMSE score get smaller and the performance gets better as the feature increase. The local validation result of FeatureSet₃ is 0.1275, which improves 0.0188 RMSE scores compared to the Baseline. The experimental results prove the effectiveness of our proposed feature engineering. 2) Comparing the base model and stacked model, the stacked model "Stacked" has 0.1267 RMSE scores, which improves by 0.0008 RMSE scores compared to the base model FeatureSet₃. The results of the comparison show that the model stacking strategy can improve the generalisation performance of the base model. Both the results of local validation and the public leaderboard can prove the effectiveness of our proposed approach.

V. CONCLUSION

In this work, we propose a gradient boosting based approach to predict the cost of forwarding contracts. We first analyse the data related to the freight forwarding contracts and provide the guidance for the feature engineering. Then, we focus on the feature engineering step to generate new features from the given data which can effectively characterise the cost of contracts. Finally, we present an ensemble approach that introduces the model stacking mechanism to improve the generalisation performance of base models. Both the results of our self-validation and the competition have shown that our

proposed approach is competitive. Future work can focus on trying a deep learning model as the base model for modelling the contract data and considering time as a trend factor for the features.

ACKNOWLEDGMENT

This research is supported by National Key Research and Development Program of China(No.2021YFF0307203, No.2019QY1300), and Youth Innovation Promotion Association CAS (No.2021156), the Strategic Priority Research Program of Chinese Academy of Sciences (No.XDC02040100). This work is also supported by the Program of Key Laboratory of Network Assessment Technology, the Chinese Academy of Sciences, Program of Beijing Key Laboratory of Network Security and Protection Technology. We thank the anonymous reviewers for their feedbacks on the paper and volunteers of the FedCSIS 2022 Challenge.

REFERENCES

- [1] P. Anitha and M. M. Patil, "Forecasting of transportation cost for logistics data," in *2021 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*. IEEE, 2021. doi: 10.1109/CONECCT52877.2021.9622576 pp. 01–06.
- [2] R. Leszczyna, "Cost of cybersecurity management," in *Cybersecurity in the Electricity Sector*. Springer, 2019, pp. 127–147.
- [3] D. Chakraborty, H. Elhegazy, H. Elzarka, and L. Gutierrez, "A novel construction cost prediction model using hybrid natural and light gradient boosting," *Advanced Engineering Informatics*, vol. 46, p. 101201, 2020. doi: 10.1016/j.aei.2020.101201
- [4] M. A. Morid, O. R. L. Sheng, K. Kawamoto, T. Ault, J. Dorius, and S. Abdelrahman, "Healthcare cost prediction: Leveraging fine-grain temporal patterns," *Journal of biomedical informatics*, vol. 91, p. 103113, 2019. doi: 10.1016/j.jbi.2019.103113
- [5] (2022, Jun.) Fedcsis 2022 challenge: Predicting the costs of forwarding contracts. [Online]. Available: <https://knowledgepit.ml/fedcsis-2022-challenge/>
- [6] A. Janusz, A. Jamiołkowski, and M. Okulewicz, "Predicting the costs of forwarding contracts: Analysis of data mining competition results," in *Proceedings of the 17th Conference on Computer Science and Intelligence Systems, FedCSIS 2022, Sofia, Bulgaria, September 4-7, 2022*. IEEE, 2022.
- [7] J. H. Friedman, "Greedy function approximation: a gradient boosting machine," *Annals of statistics*, pp. 1189–1232, 2001. doi: 10.1214/aos/1013203451
- [8] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: a statistical view of boosting," *The annals of statistics*, vol. 28, no. 2, pp. 337–407, 2000. doi: 10.1214/aos/1016218223
- [9] H. Xiao, Y. Liu, D. Du, and Z. Lu, "Wp-gbdt: An approach for winner prediction using gradient boosting decision tree," in *2021 IEEE International Conference on Big Data (Big Data)*, 2021. doi: 10.1109/BigData52589.2021.9671688 pp. 5691–5698.
- [10] S. Tyree, K. Q. Weinberger, K. Agrawal, and J. Paykin, "Parallel boosted regression trees for web search ranking," in *Proceedings of the 20th international conference on World wide web*, 2011. doi: 10.1145/1963405.1963461 pp. 387–396.
- [11] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016. doi: 10.1145/2939672.2939785 pp. 785–794.
- [12] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "Lightgbm: A highly efficient gradient boosting decision tree," *Advances in neural information processing systems*, vol. 30, pp. 3146–3154, 2017.
- [13] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorigush, and A. Gulin, "Catboost: unbiased boosting with categorical features," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, 2018, pp. 6639–6649.
- [14] Z.-H. Zhou, *Ensemble learning*. Springer, 2021, pp. 181–210.
- [15] P. Refaeilzadeh, L. Tang, and H. Liu, "Cross-validation," *Encyclopedia of database systems*, vol. 5, pp. 532–538, 2009.

Computer Science and Systems

CSS is a FedCSIS track aiming at integrating and creating synergy between FedCSIS technical sessions which thematically subscribe to more technical (or applicable) aspects of computer science and related disciplines. The CSS track spans themes ranging from hardware issues close to the discipline of computer engineering via software issues tackled by the theory and applications of computer science, and to communication issues of interest to distributed, smart, multimedia and network systems.

The track is oriented on the research where the computer science meets the real world problems, real constraints, model objectives, etc. However the scope is not limited to applications, we all know that all of them were born from the innovative theory developed in laboratory. We want to show the fusion of these two worlds. Therefore one of the goals for the track is to show how the idea is transformed into application, since the history of modern science show that most of successful research experiments had their continuation in real world. CSS track is going to give an international panel where researchers will have a chance to promote their recent advances in applied computer science both from theoretical and practical side.

SCOPE

- Applied parallel and distributed computing and systems
- Applied system architectures and paradigms
- Problem-oriented simulations and modelling
- Applied methods of multimodal, constrained and heuristic optimization
- Applied computer systems in technology, medicine, ecology, environment, economy, etc.
- Theoretical fundamentals of the above computer sciences developed into the practical use
- Hardware engineering

Track includes technical sessions:

- Actors, Agents, Assistants, Avatars (1st Workshop 4A'22)
- Computer Aspects of Numerical Algorithms (15th Workshop CANA'22)
- Concurrency, Specification and Programming (30th International Symposium CS&P'22)
- Multimedia Applications and Processing (15th International Symposium MMAP'22)
- Scalable Computing (12th Workshop WSC'22)

TRACK CHAIRS

- **Dimov, Ivan**, Bulgarian Academy of Sciences, Institute of Information and Communication Technologies, Bulgaria
relax
- **Wasielewska-Michniewska, Katarzyna**, Systems Research Institute, Polish Academy of Sciences, Poland

PROGRAM CHAIRS

- **Dimov, Ivan**, Bulgarian Academy of Sciences, Institute of Information and Communication Technologies, Bulgaria
- **Wasielewska-Michniewska, Katarzyna**, Systems Research Institute, Polish Academy of Sciences, Poland

PROGRAM COMMITTEE

- **Barbosa, Jorge**, University of Porto, Portugal
- **Braubach, Lars**, University of Hamburg, Germany
- **Cabri, Giacomo**, Università di Modena e Reggio Emilia, Italy
- **Georgiev, Krassimir**, Institute of Information and Communication Technologies, Bulgarian Academy of Sciences, Bulgaria
- **Homles, Violeta**, University of Huddersfield, United Kingdom
- **Jezic, Gordan**, University of Zagreb, Croatia
- **Lirkov, Ivan**, Institute of Information and Communication Technologies, Bulgarian Academy of Sciences, Bulgaria
- **Mangioni, Giuseppe**, Dipartimento di Ingegneria Elettrica Elettronica e Informatica (DIEEI) - University of Catania, Italy
- **Millham, Richard**, Durban University of Technology, South Africa
- **Modoni, Gianfranco**, STIIMA-CNR, Italy
- **Pandey, Rajiv**, Amity University, India
- **Petcu, Dana**, West University of Timisoara, Romania
- **Roszczyk, Radosław**, Warsaw University of Technology, Poland
- **Rycerz, Katarzyna**, AGH University of Science and Technology, Poland
- **Schreiner, Wolfgang**, Research Institute for Symbolic Computation (RISC), Austria
- **Tanwar, Sudeep**, Nirma University, Ahmedabad (Gujarat), India
- **Tudoroiu, Nicolae**, John Abbott College, Canada
- **Vardanega, Tullio**, University of Padua, Italy
- **V Vinoth Kumar**, MVJ College of Engineering, India

Block Subspace Iteration Method for Structural Analysis on Multicore Computers

Sergiy Fialko

Tadeusz Kosciuszko Cracow University of Technology
 ul. Warszawska 24 St., 31 155 Kraków, Poland
 Email: sergiy.fialko@gmail.com

Abstract—The block subspace iteration method for problems of structural dynamics oriented on multi-core computers is presented to extract the natural vibration frequencies and modes. The investigation is focused on multithreaded parallelization of all principal stages of the method allowing to determine up to several thousand eigenpairs even for design models with a lot of very close or multiple eigenfrequencies.

I. INTRODUCTION

MODERN design finite-element models of buildings and structures usually have a dense spectrum of natural vibration frequencies. Mathematically, the problem is reduced to solving a partial generalized algebraic eigenvalue problem of a large dimension

$$\mathbf{K}\mathbf{v}_i - \lambda_i \mathbf{M}\mathbf{v}_i = 0, \quad i \in [1, n]. \quad (1)$$

Here \mathbf{K} is a sparse symmetric positive definite stiffness matrix, \mathbf{M} – diagonal or sparse symmetric semi-definite mass matrix, $\{\lambda_i, \mathbf{v}_i\}$ – eigenpair, N – dimension of the problem (1), n – number of required eigenpairs.

Requirements of seismic norms to determinate such a number of eigenpairs, which will provide a sufficiently high percentage of modal masses [1], [2] for each of the seismic input directions, leads to the fact that for many design models, with dimensionality, $N = 2,000,000 \div 6,000,000$ often has to obtain several thousands of eigenpairs [3].

The vast majority of such calculations are carried out by small and medium-sized design bureaus, so the main tool for numerical solutions of such problems are multi-core computers with shared memory, on which the solution of such problems by conventional methods can take from several hours to several days. Thus, increasing the performance of numerical methods for solving problem (1) for large design models with a large number of required eigenpairs is of great importance.

Both the Lanczos method and the subspace iteration method can be seen as varieties of the Arnoldi method [4]. However, within the framework of this article, we will confine ourselves to treating these methods as a kind of inverse matrix iteration method.

In the existing FEA software for the problems of structural dynamics, various versions of the block Lanczos method with spectral transformations [5], [6], etc. are very popular. The block version of the Lanczos method allows us

to confidently solve problems for which there are a large number of multiple or almost multiple eigenfrequencies, in other words, the spectrum of eigenfrequencies has the area of condenses. Spectral transformations of type

$$\mathbf{M}\mathbf{v}_i - \frac{1}{\lambda_i - \sigma} (\mathbf{K} - \sigma\mathbf{M})\mathbf{v}_i = 0, \quad i \in [1, n], \quad (2)$$

where σ - shift, first, allow us to better separate the close eigenfrequencies, which speeds up the convergence of the method, and secondly, divide the desired frequency interval to relatively small sub-intervals, limiting the dimension of the reduced problem on the Krylov subspace even in the case when we need to determine several thousand eigenpairs. At the same time, the maximum dimension of the reduced problem on the Krylov subspace does not depend on the number of required eigenpairs, which provides a quasi-linear computational complexity of the method instead of the quadratic computational complexity typical for versions of the method operating on the single frequency interval.

With the development of multi-core computers with shared memory, it turned out that existing implementations of the Lanczos method face difficulties in multi-threaded parallelization. One of the reasons that reduce the effectiveness of parallelization is the dimensionality of Krylov's subspace changing from step to step. Our observations have shown that while the dimension of the Krylov subspace is approximately within the first third of the maximum dimension in the current frequency interval, it is not possible to effectively use all the cores of the processor.

The disadvantage of a simple version of the subspace iteration method [1] (section 14-6) is the quadratic increase in the time of solving the problem with an increase in the number of required eigenpairs, which makes it practically unacceptable for the class of problems presented here.

The block version of the shifted subspace iteration method [7] corrects many shortcomings of the previous version of the algorithm. The essence of the method is that the iterations are performed in a block of a fixed dimension, which is much smaller than the number of required eigenpairs. As soon as converged eigenpairs appear in the block, the corresponding eigenvectors are placed in the special storage, excluded from the block, and in their place are created new start vectors that

This work was supported by IT Company SCAD Soft (www.scadsoft.com)

are linearly independent of each other and orthogonalized both to the remaining vectors in the block and to the previously converged eigenvectors. At each step of the method, all the vectors in the block are orthogonalized to the previously converged eigenvectors in order to avoid duplication of eigenpairs. At each step of the method, all the vectors in the block are orthogonalized to the previously converged eigenvectors in order to avoid duplication of eigenpairs. These orthogonalization procedures are a bottleneck that limits the use of this method for problems in which several thousand eigenpairs need to be extracted.

A similar drawback is the version of the method of conjugated gradients with preconditioning and spectral transformations [3].

In this article, we present a block version of the subspace iteration method with spectral transformations as an alternative to the block Lanczos method. The dimensionality of Krylov's subspace remains constant all the time, which makes it possible to effectively use the capabilities of modern multi-core computers. Spectral transformations accelerate the convergence of the method by dividing the close natural frequencies, and most importantly, they make it possible to divide the frequency interval into computationally independent sub-intervals and provide a quasi-linear dependence of the solving time of the problem on the number of required eigenpairs.

II. PARALLEL BLOCK SUBSPACE ITERATION METHOD WITH A SPECTRAL TRANSFORMATIONS

A. Foundation

Before proceeding to the presentation of the proposed method, let us first give a simple algorithm of the subspace iteration method, corresponding to the one described in [1], but at the same time, we will take into account the impact of the shift σ . Let the approximation of the eigenvectors \mathbf{v}_i^k , forming a rectangular matrix be known at step k

$$\mathbf{Q}^k = \{\mathbf{v}_1^k, \mathbf{v}_2^k, \dots, \mathbf{v}_m^k\}, \quad (3)$$

where

$$\begin{aligned} (\mathbf{v}_i^k)^T \mathbf{M} \mathbf{v}_j^k &= \delta_{ij}, \\ (\mathbf{v}_i^k)^T (\mathbf{K} - \sigma \mathbf{M}) \mathbf{v}_j^k &= \begin{cases} 0, & i \neq j \\ \neq 0, & i = j \end{cases} \quad i, j \in [1, m], \end{aligned} \quad (4)$$

δ_{ij} is a Kronecker symbol, m – block dimension (Krylov subspaces dimension). Perform an iteration step with the inverse matrix for the expression (2):

$$(\mathbf{K} - \sigma \mathbf{M}) \overline{\mathbf{Q}}^{k+1} = \mathbf{M} \mathbf{Q}^k. \quad (5)$$

Matrix $\overline{\mathbf{Q}}^{k+1}$ contains improved approximations of eigenvectors compared to the matrix \mathbf{Q}^k , but $\mathbf{K} - \sigma \mathbf{M}$ and \mathbf{M} orthogonality of vectors $\overline{\mathbf{v}}_i^{k+1}$, $i \in [1, m]$ is lost. Therefore, the next step is to orthogonalize the column vectors of the matrix $\overline{\mathbf{Q}}^{k+1}$:

$$\mathbf{v}_i^{k+1} = \sum_{j=1}^m q_{i,j}^{k+1} \overline{\mathbf{v}}_j^{k+1}, \quad i \in [1, m]. \quad (6)$$

Substituting (6) in (2), we get an algebraic generalized eigenvalue problem in the subspace S_m formed by the vectors $\overline{\mathbf{Q}}^{k+1}$:

$$\frac{1}{\lambda_i - \sigma} (\{\mathbf{k}\} - \sigma \{\mathbf{m}\}) \mathbf{q}_i - \{\mathbf{m}\} \mathbf{q}_i = 0, \quad i \in [1, m], \quad (7)$$

where $\{\mathbf{k}\} = \mathbf{Q}^T \mathbf{K} \mathbf{Q}$, $\{\mathbf{m}\} = \mathbf{Q}^T \mathbf{M} \mathbf{Q}$, $\mathbf{q}_i = \{q_{i,1}, q_{i,2}, \dots, q_{i,m}\}^T$ is a vector of dimension m containing the expansion coefficients (6). Here, the upper subscript $k+1$ is omitted for the sake of brevity. Problem (7) is equivalent to a simpler problem:

$$\{\mathbf{k}\} \mathbf{q}_i - \lambda_i \{\mathbf{m}\} \mathbf{q}_i = 0, \quad i \in [1, m], \quad (8)$$

which is solved by using the LAPACK procedures implemented in the Intel Math Kernel Library [8]. Having determined at step $k+1$ the approximations of eigenvalues and eigenvectors in subspace S_m , we obtain approximations of eigenvectors (Ritz vectors) in the source space of dimension N :

$$\mathbf{Q}^{k+1} = \overline{\mathbf{Q}}^{k+1} \{\mathbf{q}^{k+1}\}, \quad (9)$$

where $\{\mathbf{q}\} = \{\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_m\}$. This ensures the

$\mathbf{K} - \sigma \mathbf{M}$ and \mathbf{M} orthogonality of the vectors \mathbf{v}_i^{k+1} , $i \in [1, m]$, since

$$\begin{aligned} (\mathbf{v}_i^{k+1})^T \mathbf{M} \mathbf{v}_j^{k+1} &= (\mathbf{q}_i^{k+1})^T (\mathbf{Q}^{k+1})^T \mathbf{M} \mathbf{Q}^{k+1} \mathbf{q}_j^{k+1} = \\ &= (\mathbf{q}_i^{k+1})^T \{\mathbf{m}^{k+1}\} \mathbf{q}_j^{k+1} = \delta_{ij} \end{aligned}, \quad (10)$$

and

$$\begin{aligned} (\mathbf{v}_i^{k+1})^T \mathbf{K}_\sigma \mathbf{v}_j^{k+1} &= (\mathbf{q}_i^{k+1})^T (\mathbf{Q}^{k+1})^T \mathbf{K}_\sigma \mathbf{Q}^{k+1} \mathbf{q}_j^{k+1} = \\ &= (\mathbf{q}_i^{k+1})^T \{\mathbf{K}_\sigma^{k+1}\} \mathbf{q}_j^{k+1} = \begin{cases} 0, & i \neq j \\ \neq 0, & i = j \end{cases}, \end{aligned} \quad (11)$$

where $\mathbf{K}_\sigma = \mathbf{K} - \sigma \mathbf{M}$, $\{\mathbf{k}_\sigma\} = \{\mathbf{k}\} - \sigma \{\mathbf{m}\}$.

Here, the central moment that ensures the convergence of the method is the expression (5). Stages (6) to (9) provide the orthogonality of the improved Ritz vectors and prevent duplication of eigenpairs. Let's expand the expression (5) according to the eigenvectors of the problem (2) and multiply left by \mathbf{v}_j^T :

$$\sum_{i=1}^m \alpha_i^{k+1} (\mathbf{K} - \sigma \mathbf{M}) \mathbf{v}_i = \sum_{i=1}^m \alpha_i^k \mathbf{M} \mathbf{v}_i \cdot |\mathbf{v}_j^T, \quad j \in [1, m]. \quad (12)$$

Here $\mathbf{v}_i^k = \sum_{i=1}^m \alpha_i^k \mathbf{v}_i$, \mathbf{v}_i^k is an approximation of eigenvector

\mathbf{v}_i in the iteration step k . Taking into account the orthogonality of eigenvectors and the norming conditions with respect to the matrix \mathbf{M} , and also applying the expression for Rayleigh's quantity $\lambda_i = \mathbf{v}_i^T \mathbf{K} \mathbf{v}_i$ for the problem (1), we get:

$$\alpha_i^k = \left(\frac{1}{\lambda_i - \sigma} \right)^k \alpha_i^0, \quad i \in [1, m], \quad (13)$$

from which it follows that the closer the shift σ to the eigenvalue λ_i , the faster will be the convergence of the iterative process $k = 1, 2, \dots$ to eigenpair $\{\lambda_i, \mathbf{v}_i\}$. Figure 1 presents a typical pattern of convergence of eigenvalues around the shift σ : the closer the eigenvalue λ_i is to the shift σ , the faster “on average” the Ritz pair $\{\lambda_i^k, \mathbf{v}_i^k\}$ converges to the corresponding eigenpair $\{\lambda_i, \mathbf{v}_i\}$ with the required precision.

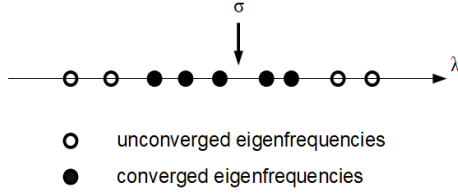


Fig. 1 Typical convergence of eigenfrequencies for methods based on shifted inverse iterations

The phrase "on average" should be understood in the statistical sense, since the rate of convergence depends not only on the difference $\lambda_i - \sigma$, but also on the selection of the initial approximation determined by α_i^0 coefficients. Thus, with the help of a proper selection of the shift value σ , we can control the position of the "center of convergence" on the axis λ . The information provided is well known and is presented here to facilitate understanding of the further presentation.

B. Block subspace iteration method

The disadvantage of the algorithm outlined above is that, with an increase in the number of required eigenpairs, the dimension m of the subspace increases, which leads to a rapid increase in the duration of such an analysis. To overcome this drawback, we divide the frequency interval into the subintervals, keeping a relatively small value of the parameter m , and also use multi-threaded parallelization. Unfortunately, the amount of RAM of modern multi-core computers does not allow us to concurrently solve the plurality of problems (2) for different shift values σ , as is done for distributed memory systems, so we parallelize the separate stages of the method.

The following Algorithm 1 presents the proposed approach. Step 1 performs the starting initialization of the method – it creates m orthogonal and \mathbf{M} – orthonormal vectors forming the rectangular matrix \mathbf{Q}^0 . *LeftMark* and *Rightmark* correspond to the left and right borders of the subinterval on the axis λ (Fig. 2), and *no_conv_modes* means the number of converged eigenpairs.

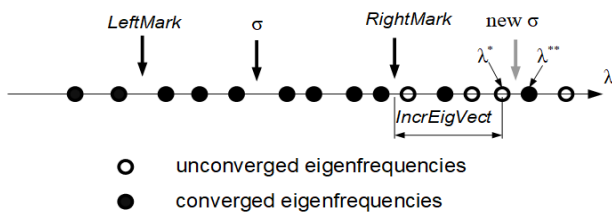


Fig. 2 Next frequency subinterval preparation

Algorithm 1. General algorithm of the block subspace iteration method with spectral transformation.

1. Creation of m linearly independent vectors \mathbf{Q}^0 according (3). Set $k = 1$; $no_conv_modes = 0$; $LeftMark = RightMark = 0$; $\sigma = 0$.
2. **while** $no_conv_modes < nModes$ **do**
3. Check status of vectors \mathbf{Q}^k in block.
4. Inverse iteration step (5).
5. Create subspace matrices $\{\mathbf{k}\}^k$ and $\{\mathbf{m}\}^k$.
6. Solve reduced eigenvalue problem (8).
7. Calculate new Ritz vectors (9).
8. $k++$.
9. **end while**

Algorithm 2. Check status of vectors \mathbf{Q}^k in block

1. **parallel for** $i = 1; i \leq m; ++i$ **do**
2. $\mathbf{r}_i^k = \mathbf{K}\mathbf{v}_i^k - \lambda_i^k \mathbf{M}\mathbf{v}_i^k$
3. **if** $\|\mathbf{r}_i^k\|_2 / \|\lambda_i^k \mathbf{M}\mathbf{v}_i^k\|_2 < tol$
4. $conv_i = true$;
5. **else**
6. $conv_i = false$;
7. **end of parallel for**
8. *SetShiftProc()*;
9. **if** $\sigma_{new} == \sigma$ **then**
10. **return**;
11. **else**
12. $\sigma = \sigma_{new}$;
13. **end if**
14. $\forall \lambda_i \in [LeftMark, RightMark]$ store $\{\lambda_i, \mathbf{v}_i\}$ as a final results. Put: $list_new_vect \leftarrow i$.
15. $LeftMark = RightMark$;
16. **Parallel for** $i \in list_new_vect$ **do**
17. Generate new start vectors instead of stored eigenvectors, orthogonalize them against all remaining vectors in the block and normalize $\mathbf{v}_i^T \mathbf{M}\mathbf{v}_i = 1$.
18. **end of parallel for**

Loop **while** (steps 2 – 9) works until the number of required eigenpairs $nModes$ are defined.

In step 3, the status of vectors in the block is checked - Algorithm 2. In a parallel loop **for** (steps 1 – 7), the residual vector \mathbf{r}_i^k is determined. If the condition of step 3 is satisfied, this means that this Ritz pair has been converged with the required precision set by the *tol* parameter, and the i^{th} element of the *conv* array is assigned to *true*.

The *SetShiftProc()* procedure (step 8) calculates the number of converged eigenpairs, starting with *LeftMark* until the first non-converged Ritz pair meets, and the position *RightMark* (Fig. 2) is determined. Thus, segment $[LeftMark, RightMark]$ contains only eigenvalues for converged

eigenpairs. The transition to a new subinterval (change in the value of the shift σ) is carried out when the following two conditions are satisfied:

- The number of converged eigenpairs on the segment $[LeftMark, \sigma]$ will be equal to $NoNegSignes - no_conv_modes$. Here, $NoNegSignes$ is the number of sign changes on the main diagonal of the factorized matrix $\mathbf{K} - \sigma\mathbf{M}$ (the number of eigenvalues enclosed in the interval from zero to σ), and no_conv_modes is the number of converging eigenpairs, which should be equal to the number of eigenvalues enclosed in the interval from zero to $LeftMark$ (Sylvester's theorem of inertia).
- The number of converged eigenpairs on the $[LeftMark, RightMark]$ segment is not less than the specified $IncrEigVect$ value.

The first condition ensures that there are no skipped eigenfrequencies during the transition from one subinterval to another, and the second is necessary to ensure a sufficient number of iterations of the method for reliable prediction of the $RightMark$ value.

As soon as the above conditions are satisfied, a transition to a new subinterval is carried out: a new shift value is calculated as $\sigma_{new} = (\lambda^* + \lambda^{**})/2$, where λ^* is an approximation of eigenvalue locating on a value $IncrEigVect$ from rightmost converged eigenvalue belonging to segment $[LeftMark, RightMark]$, and λ^{**} is the next eigenvalue – see Fig. 2. To ensure the computational stability of the proposed algorithm, the prediction λ^{**} must be made with sufficient accuracy, which depends on the dimension of the block m and the value of the $IncrEigVect$ parameter. The recommended values of these parameters are discussed in section III, E.

If at least one of the above conditions is not fulfilled, the shift value does not change, there is no transition to a new subinterval, the exit from Algorithm 2 (step 9) is performed and the iterations in this \mathbf{Q}^k block continue. Otherwise, the σ shift value is changed and the transition to a new frequency interval is made (step 12). All converged eigenpairs in the $[LeftMark, RightMark]$ segment are the final result and are placed in special storage on disk (step 14) and the $LeftMark$ value is reset (step 15). Then, in a parallel region (steps 16 – 18), new starting vectors are generated in the addresses of converged eigenvectors, orthogonalized against themselves, as well as against the remaining vectors in the block and normalized, after which we proceed to step 4 of Algorithm 1.

If at this step k there is a change in the magnitude of the shift σ , the factorization of the matrix $\mathbf{K} - \sigma\mathbf{M}$ is performed. Otherwise, the lower triangular matrix with the previous factorization is used. Forward-back substitutions are then performed. Multithreaded parallelization is used when calculating the rectangular matrix of the right parts (5) is performed. To factorize the matrix, the PARFES solver [9] designed specifically for multi-core computers with shared memory is used. Also, a parallel method [10] is applied to perform forward-back substitutions.

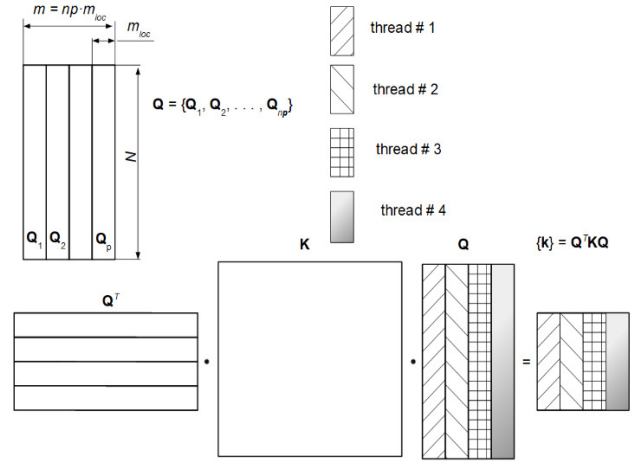


Fig. 3 Subdivision the matrix \mathbf{Q} between threads

Calculation of projection matrices $\{\mathbf{k}\}^k = (\mathbf{Q}^k)^T \mathbf{K} \mathbf{Q}^k$ and $\{\mathbf{m}\}^k = (\mathbf{Q}^k)^T \mathbf{M} \mathbf{Q}^k$ in subspace $S_m = \text{span}\{\mathbf{v}_i^k \in \mathbf{Q}^k\}$ (Algorithm 1, step 5), is produced using multithreading.

To do this, the matrix \mathbf{Q} (the iteration number k is omitted) is divided into blocks $\mathbf{Q}_1, \mathbf{Q}_2, \dots, \mathbf{Q}_{np}$, where np is the number of threads and $m_{loc} = m/np$ is the number of vectors in each block \mathbf{Q}_p , $p \in [1, np]$ (Fig. 3). If the dimension m of the iterated block \mathbf{Q} is not multiple to the number of threads, then for the last block is taken $m_{loc} = m - (np - 1)m/np$, where $(np - 1)m/np$ is the integer part of this expression. The mapping of blocks $\mathbf{Q}_1, \mathbf{Q}_2, \dots, \mathbf{Q}_{np}$ onto threads is shown in Fig. 3. Each thread performs a task:

$$\{\mathbf{k}_{ip}\} = \mathbf{Q}^T \mathbf{K} \mathbf{Q}_{ip}, \quad \{\mathbf{m}_{ip}\} = \mathbf{Q}^T \mathbf{M} \mathbf{Q}_{ip}, \quad ip \in [1, np]. \quad (14)$$

First, the sparse matrix \mathbf{K} is multiplied by a dense rectangular matrix $\mathbf{B}_{ip} = \mathbf{K} \cdot \mathbf{Q}_{ip}$, and then $\{\mathbf{k}_{ip}\} = \mathbf{Q}^T \mathbf{B}_{ip}$ is calculated. Similarly, the dense matrix $\{\mathbf{m}_{ip}\}$ is evaluated.

The generalized eigenvalue problem (8) in the S_m subspace (Algorithm 1, step 6) is solved using the LAPACK procedure of the Intel MKL library [8], after which the vectors \mathbf{v}_i^{k+1} , $i \in [1, m]$ are determined (9), using a parallel version of the $dgemm$ procedure from the Intel MKL library (Algorithm 1, step 7).

III. NUMERIC RESULTS

Numerical results have been obtained on computer with 12-core Intel® Core™ i9-9920X CPU 3.50 GHz processor, 128 GB RAM, 64-bit Windows 10 Pro OS. This processor supports SIMD instructions AVX512F and FMA, has 512-bit registers that allow loading eight double words and simultaneously perform 8 multiplications and 8 additions.

The main attention is paid to the analysis of the time of solving the problem and the acceleration of the main stages of the proposed approach with an increase in the number of threads. The comparison is made with the block Lanczos method with spectral transformations [6], developed for the SCAD FEA software [11] and using the same PARFES solver to solve systems of linear algebraic equations with sparse symmetric matrices as the proposed approach. In addition, a

multi-threaded parallelization of all the main stages of the Lanczos method was performed.

Design models are taken from the collection of problems of SCAD Soft – IT company, the developer of SCAD, one of the most widespread FEA software for the analysis and design of building structures in CIS countries, which has a certificate of compliance with regional building codes and a license for use in the design of nuclear power facilities.

A. Example 1

Figure 4 presents a design model of multistorey building comprising 4,262,958 equations. 500 eigenpairs are extracted. Such number of eigenpairs ensures the sum of modal masses 90% for each seismic input direction and satisfies requirements of the seismic building codes.

Dimension of block is accepted as $m = 96$ and parameter – $IncrEigVect = 15$. Table I depicts the total duration for the entire time of solving the problem of each basis stage of the proposed parallel block subspace iteration method (PBSI) depending on the number of threads. The number of threads does not exceed the number of physical core. This used thread binding to logical processors, where only one thread runs on each physical core. With the exclusive use of a computer for only one computing task, this strategy allows us to achieve the greatest performance and speed up the method while increasing the number of threads.

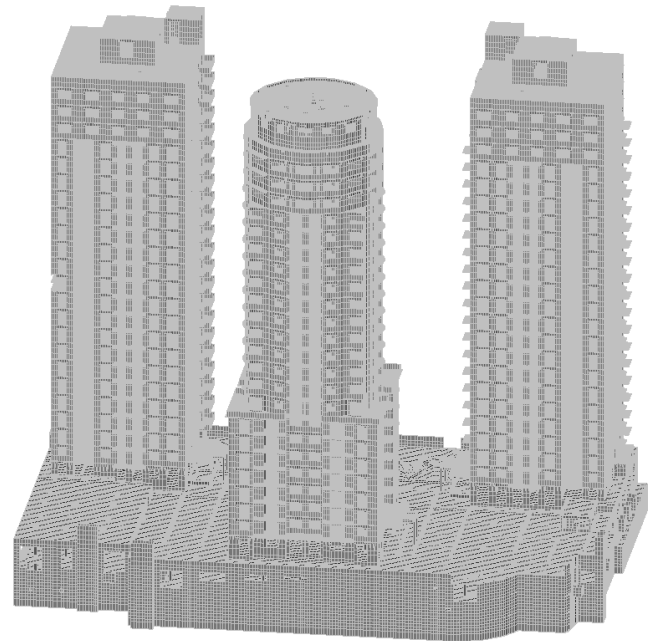


Fig. 4 Multistorey building 4,262,958 equations.

TABLE I.
DURATION OF THE BASIS STAGES OF THE PARALLEL BLOCK SUBSPACE ITERATION METHOD, S

# threads	Check status of Q^k vect.	Factorization	Resolution	Subspace matrices	Improved Ritz vectors	Rest	Total
1	1674	4744	5358	1719	284	1568	15347
2	1028	2386	3014	902	191	802	8323
4	638	1259	1770	491	123	409	4689
6	542	878	1342	365	105	298	3531
8	501	711	1159	310	101	242	3024
10	488	631	1024	275	92	203	2713
12	461	544	1013	262	98	206	2584
S_{12}	3.63	8.73	5.23	6.56	2.89	7.61	5.94

TABLE II.
DURATION OF THE BASIC STAGES OF THE BLOCK LANCZOS METHOD WITH SPECTRAL TRANSFORMATIONS, S

# threads	Generation of Lanczos vectors	Reorthogonalization	Subspace eigenvalue problem	Check of precision	Total
1	7516	3769	1230	319	13264
2	4445	2225	1238	235	8196
4	2741	1310	1227	214	5472
6	2655	1374	1218	189	5029
8	2222	945	1229	192	4480
10	2208	936	1224	180	4360
12	2194	937	1256	198	4343
S_{12}	3.43	4.02	0.98	1.61	3.05

The column «Check status of Q^k vect.» shows the duration of step 3 of Algorithm 1, the column «Factorization» – shows the duration of factorization of the sparse matrix $K - \sigma M$ by the PARFES solver [9], and the column «Resolution» depicts the time spent on performing multiplication of the mass matrix M by a rectangular matrix Q^k and on forward-back

substitutions (5). At least for the diagonal mass matrix used in this problem, the main time of this step is forward-back substitutions, since the calculation of MQ^k is performed in about 5 seconds. The column «Subspace matrices» presents the time of evaluation of matrices $\{k\}$ and $\{m\}$ (step 5, Algorithm 1). The column «Improved Ritz vectors» depicts

the time of the improved Ritz vectors evaluation \mathbf{Q}^{k+1} (step 7, Algorithm 1), the column «Rest» demonstrates the time of all remaining operations which are parallelized too, and column «Total» – total time for solution of problem. Reduced generalized eigenvalue problem on subspace S_m (step 6, Algorithm 1) is solved in sequential mode, and the total duration of this stage for the given problem is about 0.2 s.

The last row shows the speedup for 12 threads: $S_{12} = T_1/T_{12}$, where T_1 is the total time of this stage on one thread, and T_{12} is on 12 threads.

The total speedup of the PBSI method is 5.94 on 12 threads, with the highest speedup achieved at the matrix factorization stage and the lowest – at the improved Ritz vector calculation stage. Taking into account that the duration of this stage is about an order of magnitude less than the duration of forward and back substitutions, we conclude that the bottlenecks of the proposed method are the stages "Check status of \mathbf{Q}^k vect." and forward-back substitutions, having a speedup 3.63 and 5.23, respectively.

The forward and back substitution algorithms are presented in details in [10]. It also shows that the greater the number of right-hand sides, the higher the speedup of this algorithm with increasing the thread number. In the block Lanczos method [6], we usually have 7 right-hand sides and speedup does not exceed 2.5. The proposed approach has much more than 7 right-hand sides, so the speedup of forward-back substitutions is higher compared with the block Lanczos method.

Table II shows the results for the block Lanczos method with spectral transformations. The basis stages here are "Generation of Lanczos vectors", "Reorthogonalization", "Subspace eigenvalue problem", "Check of precision". Factorization of the $\mathbf{K} - \sigma\mathbf{M}$ matrix is performed with each change in the shift σ . The factorization time of the matrix and the time of forward and back substitutions are included in the total time of the "Generation of Lanczos vectors" stage. The "Subspace eigenvalue problem" stage is executed on a single thread. The total speedup of the block Lanczos method was almost two times worse than the PBSI method, and the total duration of the solution on 12 threads was 1.7 times longer.

B. Example 2

Figure 5 presents a design model of a shopping and entertainment center (TRK), containing 2,442,846 equations. Here we use an abbreviator TRK in the original (Ukrainian) language.

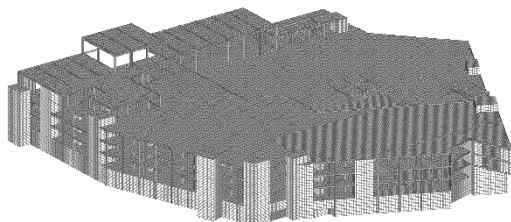


Fig. 5 Design model of a shopping and entertainment center (TRK), 2,442,846 equations.

Unlike the previous one, this design model, due to its low height, has much greater rigidity in the horizontal direction, which leads to slow convergence of the corresponding sums of modal masses. To achieve 90% of the sums of modal masses for each of the directions of seismic excitation, it was necessary to extract 2270 eigen pairs. At many parts, the natural vibration frequency spectrum undergoes condensation – Table III. In red color, especially close eigenfrequencies are highlighted.

For the PBSI method, the following parameter values are accepted: $m = 96$, $IncrEigVect = 15$. The duration of solving the problem on 12 threads is 6,729 seconds.

The time to solve this problem by the block Lanczos method is 11,883 s. Thus, the proposed PBSI method was 1.77 times faster than the block Lanczos method.

TABLE III.
FRAGMENT OF EIGENFREQUENCIES SPECTRUM

# mode	λ	$\omega = \sqrt{\lambda}$ 1/s	$f = \omega / (2\pi)$ Hz
391	1418.3925	37.6616	5.9940
392	1421.5990	37.7041	6.0008
393	1422.8605	37.7208	6.0035
394	1426.9157	37.7745	6.0120
395	1427.4229	37.7813	6.0131
396	1428.8853	37.8006	6.0162
397	1429.2057	37.8048	6.0168
398	1430.2163	37.8182	6.0190
399	1432.9059	37.8537	6.0246
400	1433.9266	37.8672	6.0268
401	1437.4244	37.9134	6.0341
402	1439.1556	37.9362	6.0377
403	1439.8368	37.9452	6.0392
404	1441.4548	37.9665	6.0426
405	1444.9049	38.0119	6.0498
406	1445.5251	38.0201	6.0511
407	1450.2098	38.0816	6.0609
408	1450.8574	38.0901	6.0622
409	1451.5816	38.0996	6.0637
410	1452.5786	38.1127	6.0658
411	1454.6621	38.1400	6.0702
412	1456.4738	38.1638	6.0740
413	1459.8107	38.2075	6.0809
414	1462.4419	38.2419	6.0864
415	1464.1232	38.2639	6.0899

C. Example 3

Figure 6 shows a design model of an industrial building comprising 1,807,218 equations. The bearing system of such a structure is the cross walls and floors (Fig. 7), which generates a huge number of local natural vibration modes, which practically do not contribute anything to the sums of modal masses both in horizontal directions and in vertical. To achieve 90% of the sum of modal masses in all directions, 20,352 eigenpairs had to be determined for this task.

For the PBSI method, the following parameter values are accepted: $m = 96$, $IncrEigVect = 15$. The duration of solving the problem on 12 threads is 45,819 s. The duration of solving the same problem by the block Lanczos method is 61,910 s.

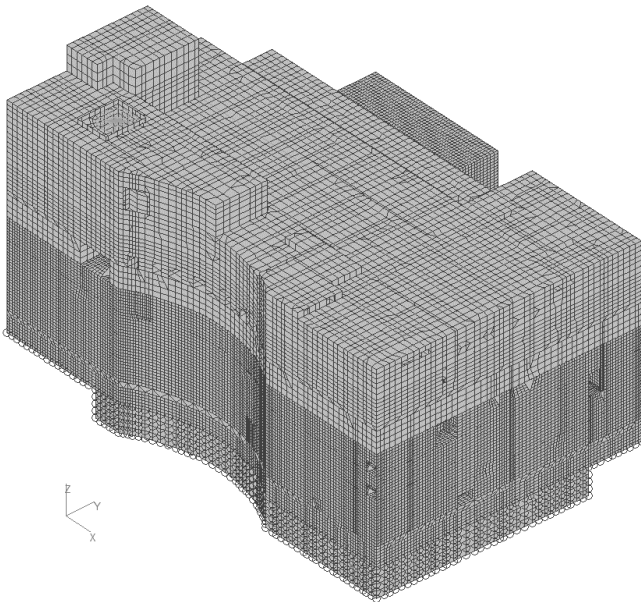


Fig. 6 Design model of industrial building – 1,807,218 equations.

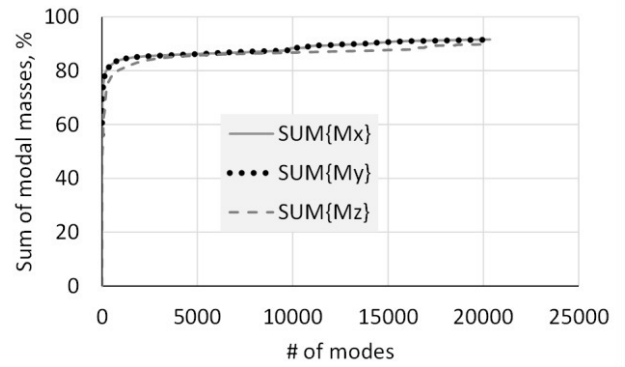


Fig. 8 Sum of modal masses in each seismic input direction

Fig. 8 shows the process of increasing the sums of modal masses with an increase in the number of eigenpairs taken into account in the modal analysis. Here $SUM\{M_x\}$, $SUM\{M_y\}$, $SUM\{M_z\}$ are the sums of modal masses in directions OX, OY, and OZ correspondingly. This problem is an excellent test for checking the computational stability and reliability of methods for determining the frequencies and modes of natural oscillations since to achieve at least 90% of the sums of modal masses in the horizontal and vertical directions, it was necessary to extract more than 20,000 eigenvalue pairs.

D. Example 4.

Design model of multistorey building is shown in Fig. 9 and Fig. 10. This model comprises 2,002,428 equations and has a complex shape of the spatial configuration.

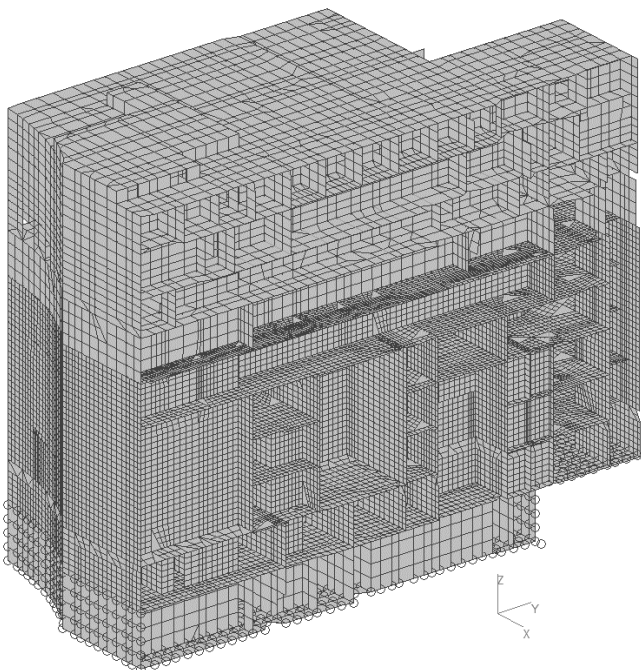


Fig. 7 Fragment of industrial building.

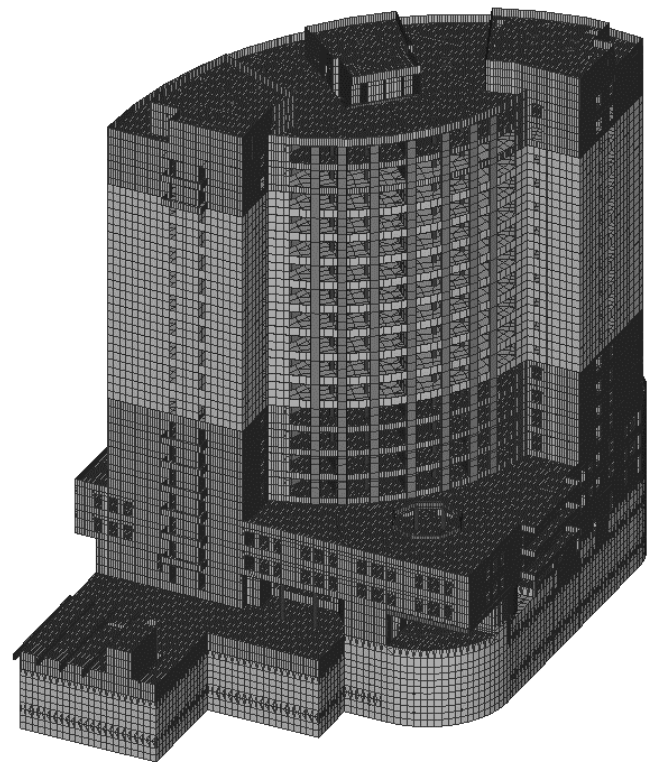


Fig. 9 Multistorey building of complex shape, 2,002,428 equations

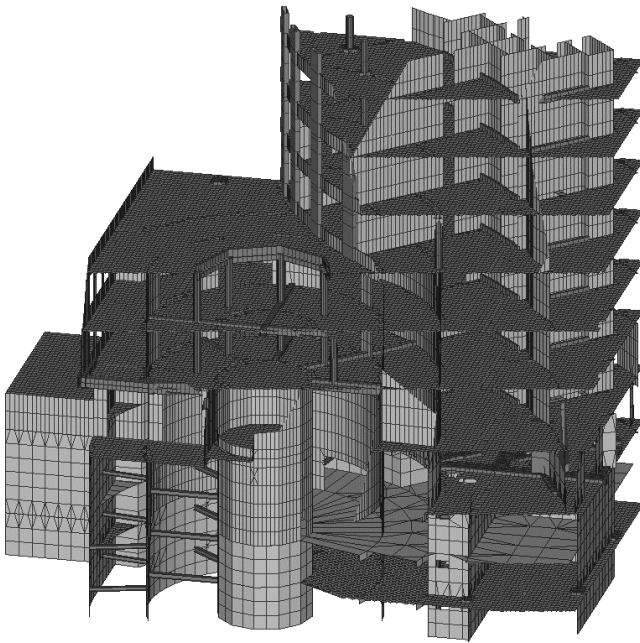


Fig. 10 Fragment of multistorey building of complex shape.

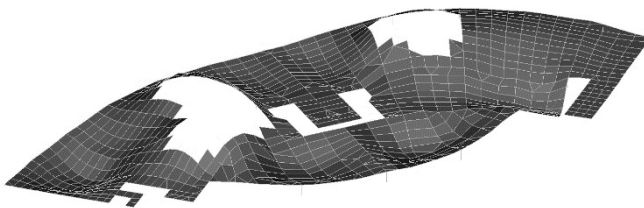


Fig. 11 Typical natural vibration mode for one from floor slabs.

A lot of local vibration modes (Fig. 11) making a very small contribution to the sum of modal masses turn out determination a large number of eigenpairs for seismic analysis – 1048 eigenpairs are required to obtain 90 % of the modal masses sum in horizontal directions and 75% in vertical (Fig. 12). This problem contains the condensation parts of the eigenvalue spectrum, one of which is presented in table IV. In red color, especially close eigenfrequencies are highlighted.

Fig. 11 demonstrates typical vertical vibrations of floor slabs. There are many such floors in the design model, so there are a large number of very close natural frequencies. In addition, the corresponding local forms of oscillation give a small contribution to the seismic response of the system, so it is necessary to determine a large number of eigenpairs to obtain reliable seismic response.

Fig. 12 demonstrates a slow increase of the sums of modal masses.

For the PBSI method, the following parameter values are accepted: $m = 96$, $IncrEigVect = 15$. The duration of solving

the problem on 12 threads is 2,687 s. The duration of solving the same problem by the block Lanczos method is 4,200 s.

TABLE IV.
FRAGMENT OF EIGENFREQUENCIES SPECTRUM

# mode	λ	$\omega = \sqrt{\lambda}$ 1/s	$f = \omega / (2\pi)$ Hz
31	2530.432	50.3034	8.006
32	2536.1223	50.3599	8.015
33	2542.0002	50.4183	8.0243
34	2547.2958	50.4707	8.0327
35	2556.1409	50.5583	8.0466
36	2560.5829	50.6022	8.0536
37	2567.1267	50.6668	8.0639
38	2567.9224	50.6747	8.0651
39	2568.2671	50.6781	8.0657
40	2575.9373	50.7537	8.0777
41	2580.8602	50.8022	8.0854
42	2582.9832	50.8231	8.0887
43	2611.8944	51.1067	8.1339
44	2634.0931	51.3234	8.1684
45	2673.8288	51.7091	8.2298
46	2700.4928	51.9663	8.2707
47	2713.0356	52.0868	8.2899
48	2721.6674	52.1696	8.3031
49	2732.2792	52.2712	8.3192
50	2752.7392	52.4666	8.3503
51	2764.7959	52.5813	8.3686
52	2784.2313	52.7658	8.3979
53	2789.7365	52.818	8.4062
54	2791.1561	52.8314	8.4084
55	2808.2844	52.9932	8.4341
56	2809.4438	53.0042	8.4359
57	2836.3514	53.2574	8.4762
58	2844.4709	53.3336	8.4883

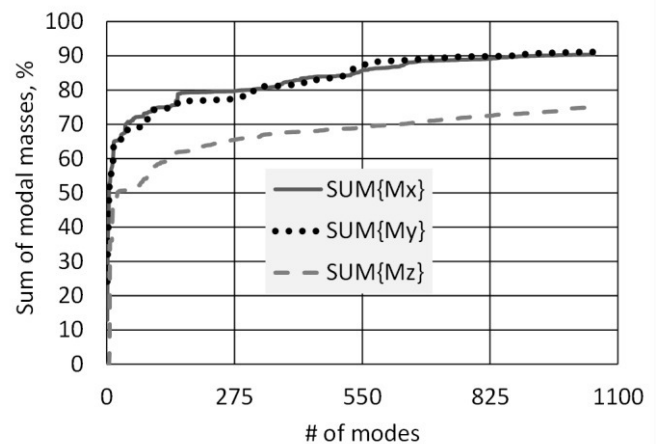


Fig. 12 Sum of modal masses in each seismic input direction

E. Reasoning about the parameters that control the convergence and numerical stability of the PBSI method.

The convergence rate and computational stability of the proposed method are determined by the values of the parameters m and $IncrEigVect$. It is desirable that the value of m be a multiple of the number of threads np . Then the number

of vectors in each subblock \mathbf{Q}_p , $p \in [1, np]$ (Fig. 3) will be the same, which will favorably affect the balance of the computational load between threads. If the values of the parameter m are too small, the Ritz pairs located to the right of *RightMark* are determined with insufficient accuracy, which often leads to an unreliable forecast when choosing a new value of the shift of the new σ – Fig. 2. In this case, it may turn out that at the new frequency interval, the number of natural frequencies of *NoNegSignes* – *no_conv_modes* enclosed between the shift value σ and *LeftMark* – significantly exceeds the *IncrEigVect*, which often leads to a loss of convergence of the iterative process. Another danger of not having enough iterations in a given frequency interval is that there will be skipped some of the natural frequencies in the [*LeftMark*, *RightMark*] segment, and then the condition that number of converged eigenpairs on the segment [*LeftMark*, σ] must be equal to *NoNegSignes* – *no_conv_modes* (see the description of the SetShiftProc() procedure) will never be fulfilled – there will be an emergency interruption of the calculations after the control number of iterations exceeds.

Too small values of the *IncrEigVect* parameter lead to insufficient accuracy in determining Ritz pairs (too few iterations in a given frequency interval), and too large – to shift the new σ to the right boundary of the interval.

Testing of a large number of different tasks from the SCAD Soft collection showed that the values of the parameters m and *IncrEigVect*, close to optimal, are as follows: $m \in [96, 192]$, *IncrEigVect* $\in [10, 30]$. In this case, lower m values correspond to smaller *IncrEigVect* values.

It should be noted that when debugging the block Lanczos method with spectral transformations [6], we encountered similar problems when choosing the values of the parameters that ensure the computational stability and convergence of the method.

IV. CONCLUSION

The considered class of problems often requires determining a large number of natural vibration frequencies and modes to satisfy to requirements of seismic codes. In addition, in many cases, due to the presence of a large number of local oscillation modes, there are areas of condensation of the natural frequency spectrum. These features lead to the fact that the considered class of problems requires the development of effective numerical methods for their solution.

The parallel block method of subspace iteration proposed in this paper, designed to solve large-scale problems of determining the natural vibration modes and frequencies of buildings, structures, and deformable solids on multi-core computers with shared memory, demonstrates a shorter analysis time and greater speedup with an increase in the number of threads than the block Lanczos method with spectral transformations [6], which has been used in many industrial software over the years.

ACKNOWLEDGMENT

The author is deeply grateful to IT company SCAD Soft for providing a collection of real-life problems with the design models created by SCAD Office users.

REFERENCES

- [1] R. W. Clough, J. Penzien, *Dynamics of Structures*. Computers and Structures Inc., Berkeley, CA, USA, 2003, pp. 623 – 638.
- [2] E.L. Wilson, *Three Dimensional Static and Dynamic Analysis of Structures*, Computers & Structures Inc., Berkeley, CA, USA, 2000, pp. 13-1 – 13-16, 17-1 – 17-8. URL: <http://www.edwilson.org/BOOK-Wilson/13-MODES.pdf>. (Accessed 20 January 2022).
- [3] S. Fialko, V. Karpilovskyi, “Block subspace projection preconditioned conjugate gradient method in modal structural analysis”, *Computers and Mathematics with Applications*, vol. 79, June 2020, pp. 3410–3428. <https://doi.org/10.1016/j.camwa.2020.02.003>.
- [4] Y. Saad, *Numerical methods for large eigenvalue problems*, in: *Classics in Applied Mathematics*, Revised ed., SIAM, Philadelphia, 2011, <http://dx.doi.org/10.1137/1.9781611970739>.
- [5] R. G. Grimes, J. G. Levis, H. D. Simon, “A shifted block Lanczos algorithm for solving sparse symmetric generalized eigenproblems”, *SIAM J. Matrix Anal. Appl.*, Vol. 15, No. 1, January 1994, pp. 228–272. Doi: DOI:10.1137/S0895479881511111.
- [6] S. Yu. Fialko, E. Z. Kriksunov, V. S. Karpilovskyi, “A block Lanczos method with spectral transformations for natural vibrations and seismic analysis of large structures in SCAD software”, in: *Proceedings of the CMM-2003 – Computer Methods in Mechanics* June 3–6, Gliwice, Poland, 2003, pp. 129–130. URL: <https://scadsoft.com/download/049P.pdf> (Accessed 20 January 2022).
- [7] E. L. Wilson, T. Itoh, “An eigensolution strategy for large systems”, *Comput. Struct.*, vol. 16, Issues 1 – 4, 1983, pp. 259–265. [https://doi.org/10.1016/0045-7949\(83\)90166-9](https://doi.org/10.1016/0045-7949(83)90166-9).
- [8] Intel MKL. URL: <https://www.intel.com/content/www/us/en/develop/documentation/onemkl-developer-reference-c/top.html> (Last access: 17.10.2021).
- [9] S. Fialko. “Parallel finite element solver for multi-core computers with shared memory”, *Computers and Mathematics with Applications*, vol. 94, 2021, pp. 1 – 14. <https://doi.org/10.1016/j.camwa.2021.04.013>.
- [10] S. Fialko. “Parallel algorithm for forward and back substitution in linear algebraic equations of finite element method”, *Journal of telecommunications and information technology*, vol. 4, 2019, pp. 20 – 29. <https://doi.org/10.26636/jtit.2019.134919>.
- [11] V. S. Karpilovskyi, E. Z. Kriksunov, A. A. Maliarenko, S. Y. Fialko, A. V. Perelmuter, M. A. Perelmuter. *SCAD Office.V.2I. System SCAD++*. Moscow: Publishing House SCAD SOFT, 2018. URL: <https://scadsoft.com/download/SCAD1033.pdf>.

Flexible user query order for the speculative query support in RDBMS.

Anna Sasak-Okon

Maria Skłodowska-Curie University in Lublin
 Pl. M. Curie-Skłodowskiej 5, 20-031 Lublin, Poland
 Email: anna.sasak@umcs.pl

Abstract—This paper concerns speculative query execution support for RDBMS based on the dynamic analysis of input (user) query stream. A middleware called the Speculative Layer is presented. Based on a specific multigraph representation of groups of consecutive input queries, called the Speculation Window, the Speculative Layer generates speculative queries for look-ahead execution. These speculatively obtained results are then used while executing user queries. This paper shortly presents the structure of the Speculative Layer and the adopted graph modelling method. Then, a new strategy of queries in the Speculation Window is introduced. Depending on the availability of executed speculative queries results we allow order of user queries in the Speculation Window changes. If a user query was to be executed without the speculative support, we prefer to delay its execution in favour of one of the consecutive user queries, expecting that speculative results obtained in the nearest future will be useful for the delayed query. The experimental results presented in a multithreaded environment, cooperating with a SQLite database, show that the proposed strategy reduces the number of user queries executed without the speculative results. Additional series of experiments verifies that the certain parameters describing the speculative support system, like Speculation Window size, are properly chosen.

I. INTRODUCTION

SPECULATIVE Parallelization, sometimes called Optimistic Parallelization[1] is a technique which allows parallel execution of code fragments that were originally intended for a sequential run. To ensure correctness, speculative results must be verified to avoid dependence violations. In such case a violating thread and its results are usually discarded and restarted with updated data [7][8][9]. The practical use of the speculative approach in relational databases usually supports single queries, ranked queries [14] or transactions (speculative transaction protocols) [12][13] by performing some anticipated, potentially useful operations or subqueries out of its standard order [10] or in advance based on received earlier tips [11]. There are also papers, like [15][16], which focus on multiple query optimization although without the concept of the speculative execution.

Graph structures often used as a formalism helping represent and analyse queries are very popular as they naturally represent and model entities and their relationships [17][18].

It should be noted that none of the described above optimization methods aims at speculative support which covers need of many future queries at a time. Saving the results obtained speculatively reminds caching methods [19][20][21][22], but

instead of history based methods we prefer to analyse queries which are waiting for execution in the nearest future and support them with data prepared in advance. The speculative execution model we introduced in our previous papers [2][3][5][6] focuses on parallelised speculative support for execution of input (user) query streams. It includes a multi-threaded middleware called the Speculative Layer, which is situated between user applications and the RDBMS. The aim of the Speculative Layer is to choose and execute speculative queries. These speculative results stored in the main memory structures called Speculative DB constitute quick access data set available while executing user queries. The process of choosing speculative queries to be executed is called the Speculative Analysis. The Speculative Analysis is performed continuously for groups of consecutive user queries called the Speculation Window (SW). For each Speculation Window a specific graph representations of each user query are created, which are then combined into one multigraph according to the defined set of rules.

The Speculation Window (SW) moves over the user query queue by one query. So far, for each SW, the first in order user query was executed nonspeculatively with or without the use of speculative results. In this paper we propose a new strategy for a nonspeculative query execution which is dependent on the availability of the speculative results. If a first query in the SW would have to be executed without speculative results we allow to replace it with a next query in the SW which has at least one executed speculative query assigned. We assume that one of the speculative results obtained in the nearest future will be useful for the delayed query. Speculation Window moves after the execution of the nonseculative query and we repeat the attempt to execute the first user query nonspeculatively.

The remaining text of the paper is composed of 3 parts. In the first part a general structure of the Speculative Layer is presented. We describe rules for query graphs creation and the process of Speculative Analysis resulting in optimally defined set of speculative queries ready for use. Section III describes the new strategy for the nonspeculative query execution. Section IV contains results of two series of experiments. First series of experiments presents effectiveness comparison between the Speculative Layer execution with old and new strategies for the nonspeculative query execution. The second series of experiments verifies the validity of the parameter

describing the size of the Speculation Window chosen for the previous versions of the speculative algorithm.

II. THE SPECULATIVE LAYER

Fast evolution of online activities induced development of database applications with the specific characteristics. These applications are intensively used as products browsing tools and thus execute many queries of a specific structure. Such queries are created by shop users who define search criteria, directly influencing attributes and conditions in the SELECT and WHERE query clauses. Therefore, the consecutive queries contain some common constituent operations, whose results, if executed speculatively can be used many times.

The above observations were an inspiration to propose a dynamic speculative support for execution of sql user queries. This model is implemented as an additional multithreaded middleware called the Speculative Layer and is located between user database applications and the RDBMS. The stream of user queries forms a queue which is continuously analysed by the Speculative Layer. The analysis (Speculative Analysis) is performed for the consecutive user queries grouped in a structure called the Speculation Window (SW) which slides on the query stream. Each user query is represented by its query graph created with a set of defined rules. Then, the single query representations for each Speculation Window are merged together to create a joint representation of the whole query group in the form of query Multigraph (Q_M).

The Speculative Layer is implemented to support and analyse CQAC (Conjunctive Queries with Arithmetic Comparisons) queries with functionality extended by two more allowed operators: IN for value sets and LIKE for strings comparisons. Additionally we allow a nested query in a WHERE clause returning a value for the attribute condition (...WHERE...attribute operator (SELECT...FROM...WHERE...)).

The aim of the Speculative Analysis process is to identify some common constituent operations in user queries which are then used to generate new queries, called the Speculative Queries. For this, an extended version of Q_M is created called Speculative Query Multigraph (SQ_M). It contains speculative edges which are a special type of edges which mark potential speculative queries. Based on the speculation result we introduce two types of speculative queries: **Speculative Parameter** or **Speculative Data** Queries. As there is allowed possibility of the modifying query occurrence in the query stream, and thus in the Speculation Window, we introduce additional type of speculative edges called **Speculative State**. Details of the strategy for modifying query handling process can be found in [6]. Fig.1 presents the SQ_M created for the following component queries:

- (Q_1) SELECT $A_{1,2}, A_{1,3}$ FROM R_1 WHERE $A_{1,2} = 4$
- (Q_2) SELECT $A_{2,2}, A_{2,3}, A_{1,3}$ FROM R_1, R_2 WHERE $A_{1,4} = A_{2,1}$ AND $A_{2,2} > 7$
- (Q_3) SELECT $A_{1,2}, A_{2,2}$ FROM R_1, R_2 WHERE $A_{1,4} = A_{1,2}$ AND $A_{1,3}$ LIKE '%xx' AND $A_{2,2} < 2$
- (Q_4) SELECT $A_{2,2}, A_{2,3}$ FROM R_2 WHERE $A_{2,2} > 5$

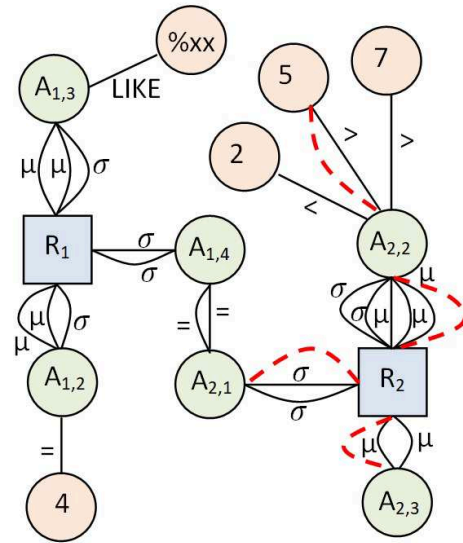


Fig. 1. Query Multigraph representing four user queries.

It has also Speculative Data edges inserted (red striped lines) which represent one of 5 possible speculative queries to be executed for this SW: (sq_1) SELECT $A_{2,1}, A_{2,2}, A_{2,3}$ FROM R_2 WHERE $A_{2,2} > 5$. Such speculative query results, if executed, can be used while executing two user queries from this SW: Q_2 and Q_4 .

Chosen speculative queries are executed in parallel with the original user queries and its results are stored in a dedicated RAM memory database structure called the Speculative DB. The data from the Speculative DB is ready to be used during execution of input user queries, called nonspeculative queries, minimizing disk database reads.

The original user query executed for each Speculation Window (so far it was always the first query in the SW) is called the nonspeculative query. If there are executed speculative queries assigned to the nonspeculative query it is transformed to use them, otherwise it is executed without the speculative support. Each user query can be executed with the use of the speculative results prepared for each relation present in its FROM clause. More details about algorithms implemented for query graph manipulation and strategies of multiple speculative queries results combining for the execution of one user query can be found in [4] and [3]. After the nonspeculative query's results are returned to the user, the SW moves by one query. As a result, the representation of the executed user query in the Q_M is replaced by the representation of the next user query from the queue, which just entered the Speculation Window. The SW move is followed by the aforementioned Speculative Analysis process to generate a new group of speculative queries for execution. Described operational scheme is repeated until there are user queries waiting for execution.

Based on previous experiments presented in [2] the Speculation Window size was set to 5 and the number of active threads for each SW was set to 3 (bigger SW or more

threads didn't provide further improvement in user queries execution). One thread is always dedicated to the execution of the nonspeculative user query, so the remaining two threads can execute chosen Speculative Queries. From the group of awaiting Speculative Queries generated in the process of the Q_M Speculative Analysis, two of them can be chosen for execution. The highest execution priority is always assigned to speculative queries which can be used by the highest number of user queries. Additionally, we consider values of two defined size reduction metrics for speculative queries - Vertical and Horizontal Selectivity (for definition see [5]). As we want to avoid creating full copies of database relations in the RAM memory we look for speculative queries with possibly low values of these metrics. If it was not possible for a particular Speculation Window to choose a new speculative query for execution, the speculative thread would report a nojob status for this SW. Executed Speculative Queries are registered on the list and assigned to user queries which can then use their results.

III. A NEW STRATEGY FOR THE SW EXECUTION

The old execution strategy for the Speculation Window (SW) was fixed, i.e. the nonspeculative query was always the first query in the SW. This way, we kept the execution order of user queries risking that for some of them the speculative results might not be ready yet.

New strategy allows the nonspeculative query not to be the first one in the SW. If the first query in the SW would be executed without the use of the speculative results, then a next query in the SW which can be executed with already obtained speculative query results is executed nonspeculatively. This way, the speculative algorithm has a chance to prepare and execute new speculative query/queries whose results will be beneficial for the delayed user query. The expected execution time reduction provided by the use of the speculative results should outweigh the potential execution delay.

Such situation is presented in Fig.2. We can see a Speculation Window containing 5 user queries (Q_2-Q_6). Below each blue user query square, there are orange circles containing ids of the executed speculative queries assigned to a particular user query for the potential use. The first query in the SW has no executed speculative queries assigned (no circles below). Thus, we look for the next query in the SW which could be executed with the use of the speculative results. The Q_3 user query is then executed nonspeculatively with the use of one of two assigned speculative queries (in certain cases it is possible to use more than one speculative result for one user query). Then SW moves, the nonexecuted Q_2 remains in it, while the Q_3 is replaced by the next query from the user query queue.

IV. EXPERIMENTAL RESULTS

A. Test Environment and Queries

The Speculative Layer is implemented in C++ with the multithreaded execution with the Pthread library. The SQLite 3.8.11.1 engine is used as a database management system. For the experiments we used the database structure and

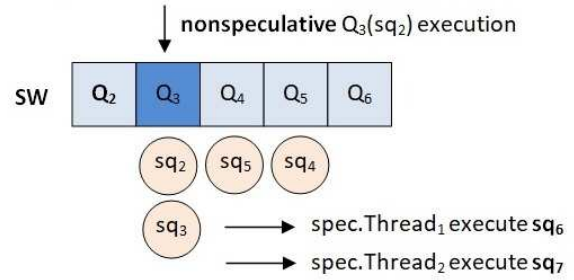


Fig. 2. New nonspeculative query execution strategy.

data (8 relations, 1GB data) from the well known database benchmark TPC-H [23]. Additionally, a set of 8 Query Templates was prepared and used to generate 3 sets of 1000 input queries each, with random values used in the attribute conditions in WHERE clauses. Structures of the T1-T8 templates are listed below with the following notation: TemplateId:RelationName(number of WHERE conditions referring to its attributes,...,RelationName(number of WHERE conditions...))

T1: LINEITEM(1 with a nested query)

T2: LINEITEM(4), PART(2)

T3: PART(3), PARTSUPP(1)

T4: LINEITEM(4), ORDERS(3), CUSTOMER(1)

T5: LINEITEM(3), PART(4), PARTSUPP(1)

T6: LINEITEM(3), ORDERS(1), CUSTOMER(1), PART(2)

T7: LINEITEM(3), ORDERS(1), CUSTOMER(1), PART(2), PARTSUPP(1)

T8: UPDATE ORDERS

Templates T2-T7 are Select query templates which join from two to five database relations. Template T1 refers to one relation but includes a nested query in its WHERE clause. T8 represents modifying queries. Each test queries set contains approximately 4% of modifying queries and 96% of select queries with the same density for each of T1-T7 templates.

B. A New Nonspeculative Query Execution Strategy

First, we have compared how the new nonspeculative query execution strategy for the Speculation Window affects the execution of user queries. For this, we compare the results obtained with the old execution strategy (nonspeculative query is always the first one in the SW) with the new strategy (nonspeculative query is the first query in the SW which can use already obtained speculative query results). The experiment was conducted for the Speculation Window size=5 and 3 active threads. Fig.3 presents average execution times obtained for each query template depending on how many speculative query results were used (from 0-red bars to 3-yellow bars speculative queries results for one user query). We can see that each speculative query used, provides further reduction in the user query execution time. The highest speedups are obtained for T5 and T7 templates. These user queries have considerably longer execution times if executed without the speculative support (approximately 163 and 181 sec.). With

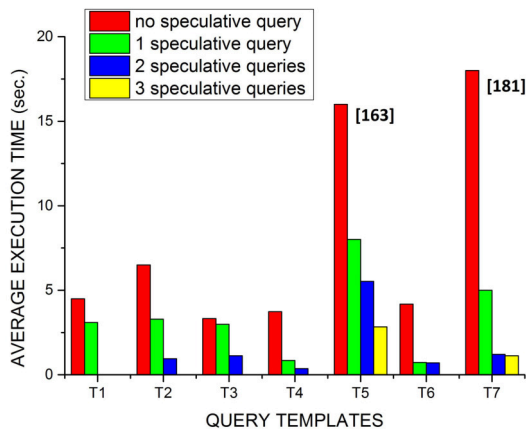


Fig. 3. The average execution time for 7 query templates with and without use of the speculative query results.

three speculative queries results used (one speculative query for each relation appearing in a query) we managed to significantly reduce their execution times (up to 9 times shorter). For the remaining templates the average time reduction is approximately 2 times. We have next studied how the new strategy influenced the general numbers of queries executed with or without the speculative support. The new strategy reduced the number of user queries executed without the use of the speculative result from approximately 113 to 57 (for 1000 user queries), which is almost two times less. The query order change (id est. as a nonspeculative query was executed a user query not being the first in the SW), was reported 132 times and concerned 58 user queries. The maximal delay of the first query in the Speculation Window execution was 4, with average delay for the whole set equal 2.59.

C. Speculation Window Size

As the implementation of the new strategy for the nonspeculative query choice provided considerable reduction in the number of user queries executed without the speculative support, we decided to run a new series of experiments to verify, if the Speculation Window (SW) size change can provide any further benefits. Table 1 presents the results we obtained for the SW sizes from 5 to 15. The size of the SW equals the number of user queries used to form a Query Multigraph. We can see that with the growing size of the SW, the number of nojob reports is also growing significantly, up to 232 for the SW size=15. The reason for this is that with bigger Speculation Windows (and bigger multigraphs) sizes, each SW move is still replacing only one user query in the multigraph structure. Such structure change is not enough to generate new and unique speculative queries for execution. Thus we can see, that the total number of executed speculative queries for the test set is actually decreasing. Even though each executed speculative query is assigned to more user queries for potential use (on average 1.98 for SW size 5 to 3.56 for SW size 15) it doesn't influence (preferably reduce) the number of user

queries executed without the speculative support. The reduced number of queries obtained for SW=5 decreases further for SW size 6 to reach an average value around 33 for bigger Speculation Windows.

Four bottom rows of Table 1 present results characteristic for the new strategy of nonspeculative query execution. We can see that the number of the user queries whose execution were delayed due to missing speculative query results, varies around 60 and doesn't seem to be dependent on the SW size. However, the bigger SW gets, the execution delay of such queries grows up to 13 steps for the SW size 15, which is not an advantageous feature. Too long delays would require a new mechanism preventing query starvation, when the waiting time predominate the potential benefits from using the speculative query results.

D. Modifying Queries

In Section IV(A), it was said that the query test sets contained approximately 4% of modifying queries. Since the modifying query enters the Speculation Window until it reaches its head and is executed as a nonspeculative query, it endangers all speculative queries to be executed on invalid data. Just after the execution of a modifying query a validation process is executed for the queries marked with the Speculative State. The aim of that process is to save (positive validation) as many speculative results as possible from being deleted. For the execution of the whole query test set with the old strategy an approximate number of 40 speculative results (for 1000 user queries) were saved from being deleted and 22 had to be deleted due to negative validation. Experiments show, that the new strategy for the nonspeculative query execution doesn't interfere with the strategy for the modifying query handling presented in [6]. For the SW size 5 numbers of deleted and saved queries remains almost unchanged. With the growing size of the Speculation Window the number of deleted speculative results grows slowly to reach an approximate value of 36 queries for the SW size 15. The number of queries endangered with invalid data and saved from being deleted grows faster and reaches an approximate value of 130 for the biggest SW, which in general doesn't seem to have a negative influence on the speculative execution.

V. CONCLUSION

The paper describes a new strategy for the nonspeculative query execution in the speculative query execution support system called the Speculative Layer. We now allow to change the order of user queries execution, if the first query in the Speculation Window would have to be executed without the use of the speculative results. The first series of experimental results for the test database and three sets of 1000 input queries each, reduced the number of user queries executed without the use of the speculative results by approximately 50%, thus in general, almost 95% of user queries were executed with the use of at least one speculative query results. The proposed speculative support provides the query average execution time reduction up to 9 times for long running queries

TABLE I
EXECUTION RESULTS OBTAINED FOR DIFFERENT SIZES OF THE SPECULATION WINDOW

	Speculation Window Size										
	5	6	7	8	9	10	11	12	13	14	15
Nojob reports	70	85	106	135	142	169	200	186	185	200	232
No Spec. Results Used	57	29	31	35	35	34	39	36	34	32	36
Total Num. of Executed Spec. Queries	1748	1717	1667	1594	1592	1539	1496	1480	1481	1445	1419
Avg. Num. of Spec. Queries Assignment	1.98	2.14	2.25	2.47	2.66	2.81	2.9	3.12	3.29	3.46	3.56
Num.of Nonspec. Queries executed with changed order	58	61	61	66	66	70	70	66	64	59	66
Num.of User Queries order change	132	180	226	274	300	335	372	383	406	373	402
Max delay in Nonspec. Query execution	4	5	6	7	8	9	11	11	12	13	13
Avg. delay in Nonspec. Query execution	2.59	2.95	3.70	4.15	4.55	4.78	5.46	5.80	6.34	6.32	6.09

(approximately 2 times for the whole test set). The execution order change was reported for approximately 58 queries, with the maximal execution delay of 4 queries (average delay 2.59). With the second series of experiments we have proved that the SW size equal 5 is a good choice. Changes in the SW size provided minor changes in the number of queries executed with the use of the speculative results. On the other hand, the average delay of a nonspeculative query execution, grew for the bigger SWs from 2.59 to 6.09.

Further work will focus on the development of an even more flexible user query execution strategy for a Speculation Window. A concept of a Variable Shift Speculation Window will be considered, which includes more than one user query executed as a nonspeculative query for each SW, depending on the available speculative results.

REFERENCES

- [1] A. Estebanez, D.R.Llanos, A.Gonzales-Escribano, "A Survey on Thread-Level Speculation Techniques," *ACM Computing Surveys*, vol. 49(2), pp. 1-39, 2017, <https://doi.org/10.1145/2938369>
- [2] A. Sasak-Okoń, "Speculative query execution in Relational databases with Graph Modelling," in *Proceedings of the FEDCSIS 2016*, ACSIS, Vol. 8., pp.1383-1387, 2016, <https://doi.org/10.15439/2016F123>
- [3] A. Sasak-Okoń, M.Tudruj, "Graph-Based speculative Query Execution in Relational Databases," in *ISPDC 2017*, Innsbruck, Austria, IEEE Explore, <https://doi.org/10.1109/ISPDC.2017.14>
- [4] A. Sasak-Okoń, M. Tudruj, "Graph-Based Speculative Query Execution for RDBMS," in *PPAM 2017*, LNCS, Vol. 10777, pp. 303-313, https://doi.org/10.1007/978-3-319-78024-5_27
- [5] A. Sasak-Okoń, M. Tudruj, "Speculative Query Execution in RDBMS Bsed in Analysis of Query Stream Multigraphs," in *24th IDEAS 2020*, Seoul, Korea, pp. 208-218, <https://doi.org/10.1145/3410566.3410604>
- [6] A. Sasak-Okoń, "Modifying Queries Strategy for Graph-Based Speculative Query Execution for RDBMS," in *PPAM 2019*, LNCS, Vol. 12043, pp. 408-418, 2020, https://doi.org/10.1007/978-3-030-43229-4_35
- [7] J. Silc, T. Ungerer, B. Robic, "Dynamic branch prediction and control speculation," *Int. Journal of High Performance Systems Arch.*, Vol. 1(1), pp.2-13, 2007, <https://doi.org/10.1504/IJHPSA.2007.013287>
- [8] S. Pan, K. So, J. T. Rahmeh, "Improving the accuracy of dynamic branch prediction using branch correlation," in *Int. Conference on Architectural Support for Programming Languages and Operating Systems*, Boston, 1992, pp.76-84, <https://doi.org/10.1145/143371.143490>
- [9] A. Moshovos, S. E. Breach, T. N. Vijaykumar, G. S. Sohi, "Dynamic Speculation and Synchronization of Data Dependences," in *24th ISCA, ACM SIGARCH Computer Architecture News*, 1997, Vol.25(2), <https://doi.org/10.1145/264107.264189>
- [10] N. Polyzotis, Y.Ioannidis, "Speculative query processing," *CIDR Conference Proceedings*, Asilomar, 2003, pp. 1-12.
- [11] G. Barish, C.A. Knoblock, "Speculative Plan Execution for Information Gathering," *Artificial Intelligence*, 2008, vol. 172(4-5), pp. 413-453, <https://doi.org/10.1016/j.artint.2007.08.002>
- [12] P.K. Reddy, M. Kitsuregawa, "Speculative locking Protocols to Improve Performance for Distributed Database Systems," *IEEE Transactions on Knowledge and Data Engineering*, 2004, Vol.16(2), p.154-169, <https://doi.org/10.1109/TKDE.2004.1269595>
- [13] T. Raganathan T, R.P. Krishna, "Improving the performance of Read-only Transactions through Asynchronous Speculation," *SpringSim Conference Proceedings*, Ottawa, 2008, p.467-474
- [14] V. Hristidis, Y. Papakonstantinou, "Algorithms and Applications for answering Ranked Queries using Ranked Views," *VLDB Journal*, 2004, Vol.13(1), p.49-70.
- [15] X.Ge, B.Yao, M.Guo, et al., "LSShare: an efficient multiple query optimization system in the cloud," *Distrib. Parallel Databases*, 2014, Vol.32(4), pp. 593-605, <https://doi.org/10.1007/s10619-014-7150-1>
- [16] M.B.Chaudhari, S.W.Dietrich, "Detecting common subexpressions for multiple query optimization over loosely-coupled heterogeneous data sources," *Distrib. Parallel Databases*, 2016, Vol.34, pp.119-143, <https://doi.org/10.1007/s10619-014-7166-6>
- [17] G.Preti, M.Lissandrini, D.Mottin, Y.Velegrakis, "Mining patterns in graphs with multiple weights," *Distributed and Parallel Databases, Special Issue on extending Database Technology*, 2019, pp.1-39, <https://doi.org/10.1007/s10619-019-07259-w>
- [18] O.Goonetilleke, D.Koutra, K.Liao, T.Sellis, "On effective and efficient graph edge labeling," *Distributed and Parallel Databases*, 2019, Vol.37, pp.5-38, <https://doi.org/10.1007/s10619-018-7234-4>
- [19] H.M. Faisal, M.A. Tariq, Atta-ur-Rahman, A. Alghamdi, N. Alowain, "A Query Matching Approach for Object Relational Databases Over Semantic Cache," *Chapter in Application of Decision Science in Business and Management*, 2020, <https://doi.org/10.5772/intechopen.90004>
- [20] M. Ahmad, M. A. Qadir, M. Sanaullah, "Query Processing Over Relational Databases with Semantic Cache: A Survey," *2008 IEEE International Multitopic Conference, Karachi*, 2008, pp. 558-564, <https://doi.org/10.1109/INMIC.2008.477801>.
- [21] F.Wang, G. Agrawal, "Query Reuse Based Query Planning for Searches over the Deep Web," *Database and Expert Systems Applications. DEXA 2010*. LNCS, Vol 6262, 2010, https://doi.org/10.1007/978-3-642-15251-1_5
- [22] P. Cybula, K. Subieta, "Query Optimization by Result Caching in the Stack-Based Approach," *Objects and Databases. ICOODB 2010*, LNCS, Vol.6348, 2010, https://doi.org/10.1007/978-3-642-16092-9_7
- [23] TPC benchmarks, <http://www.tpc.org/tpch/default.asp>, 2020.

15th Workshop on Computer Aspects of Numerical Algorithms

NUMERICAL algorithms are widely used by scientists engaged in various areas. There is a special need of highly efficient and easy-to-use scalable tools for solving large scale problems. The workshop is devoted to numerical algorithms with the particular attention to the latest scientific trends in this area and to problems related to implementation of libraries of efficient numerical algorithms. The goal of the workshop is meeting of researchers from various institutes and exchanging of their experience, and integrations of scientific centers.

TOPICS

- Parallel numerical algorithms
- Novel data formats for dense and sparse matrices
- Libraries for numerical computations
- Numerical algorithms testing and benchmarking
- Analysis of rounding errors of numerical algorithms
- Languages, tools and environments for programming numerical algorithms
- Numerical algorithms on coprocessors (GPU, Intel Xeon Phi, etc.)
- Paradigms of programming numerical algorithms
- Contemporary computer architectures
- Heterogeneous numerical algorithms
- Applications of numerical algorithms in science and technology

TECHNICAL SESSION CHAIRS

- **Bylina, Beata**, Maria Curie-Skłodowska University, Poland
- **Bylina, Jaroslaw**, Maria Curie-Skłodowska University, Poland
- **Stpiczynski, Przemyslaw**, Maria Curie-Skłodowska University, Poland

PROGRAM COMMITTEE

- **Anastassi, Zacharias**, ASPETE School of Pedagogical and Technological Education, Greece
- **Brugnano, Luigi**, Università di Firenze, Italy
- **Burczynski, Tadeusz**, Polish Academy of Sciences, Poland
- **Czachórski, Tadeusz**, Institute of Theoretical and Applied Informatics, Polish Academy of Sciences, Gliwice, Poland

- **Czarnul, Pawel**, Faculty of Electronics, Telecommunications and Informatics, Gdansk University of Technology, Poland
- **Domanska, Joanna**, Institute of Theoretical and Applied Informatics, Polish Academy of Sciences, Poland
- **Fialko, Sergiy**, Cracow University of Technology, Poland
- **Georgiev, Krassimir**, Institute of Information and Communication Technologies, Bulgarian Academy of Sciences, Bulgaria
- **Gepner, Paweł**, PAWEŁ GEPNER AI, Poland
- **Giannoutakis, Konstantinos**, Department of Applied Informatics, University of Macedonia, Greece
- **Grochla, Krzysztof**, Institute of Theoretical and Applied Informatics of PAS, Poland
- **Kozielski, Stanislaw**, Institute of Informatics, Silesian University of Technology, Poland
- **Laccetti, Giuliano**, University of Naples Federico II and INFN, Italy
- **Lirkov, Ivan**, Institute of Information and Communication Technologies, Bulgarian Academy of Sciences, Bulgaria
- **Luszczek, Piotr**, University of Tennessee Knoxville, USA
- **Marowka, Ami**, Bar-Ilan University, Israel
- **Mehmood, Rashid**, King Abdulaziz University, Saudi Arabia
- **Mrozek, Dariusz**, Silesian University of Technology, Institute of Informatics, Poland
- **Petcu, Dana**, West University of Timisoara, Romania
- **Rojek, Krzysztof**, Czestochowa University of Technology, Poland
- **Sawerwain, Marek**, University of Zielona Góra, Poland
- **Shaska, Tony**, Oakland University, USA
- **Sidje, Roger B.**, University of Alabama, USA
- **Siminski, Krzysztof**, Silesian University of Technology, Poland
- **Skubalska-Rafajłowicz, Ewa**, Wrocław University of Science and Technology, Poland
- **Szyld, Daniel**, Temple University, USA
- **Telek, Miklos**, Budapest University of Technology and Economics, Hungary
- **Tomov, Stanimire**, University of Tennessee Knoxville, USA
- **Ustimenko, Vasyl**, University of Maria Curie Skłodowska in Lublin, Poland

Automatic code optimization for computing the McCaskill partition functions

Włodzimierz Bielecki, Marek Palkowski, Maciej Poliwoła
 West Pomeranian University of Technology in Szczecin
 ul. Zolnierska 49, 71-210 Szczecin, Poland
 Email: mpalkowski@wi.zut.edu.pl

Abstract—In this paper, we present the application of three automatic source-to-source compilers to code implementing McCaskill’s bioinformatics algorithm. It computes probabilities of various substructures for RNA prediction. McCaskill’s algorithm is compute and data intensive and it is within dynamic programming. A corresponding programming code exposes non-uniform dependences that complicate tiling of that code. The corresponding code is represented within the polyhedral model. Its optimization is still a challenging task for optimizing compilers employing multi-threaded loop tiling. To generate optimized code, we used the popular PLuTo compiler that finds and applies affine transformations, the TRACO compiler based on calculating the transitive closure of loop dependence graphs, and the newest polyhedral tool DAPT implementing space-time tiling. An experimental study fulfilled on two multi-core machines: an AMD Epyc with 64 threads and a 2x Intel Xeon Platinum 9242 with 192 threads demonstrates considerable speedup, high locality, and scalability for various problem sizes and the number of threads of generated codes by means of space-time tiling.

I. INTRODUCTION

McCASKILL’S algorithm is an efficient dynamic programming one to return the value of the computed partition function $Z = \sum_P \exp(-E(P)/RT)$, where P represents all possible nested structures formed by a given RNA sequence S , $E(P)$ is the energy of structure P , R is the gas constant, and T represents temperature [1].

The approach computes the structure probabilities of each individual base pair in the RNA sequence. These probabilities are used for simultaneous folding and alignment in algorithms to predict an RNA structure with a maximum expected accuracy (MEA) [2].

Each base pair of a structure contributes a fixed energy term E_{bp} independent of its context. Under such an assumption, a partition function for a sub-sequence from position i to position j is represented with table $Q_{i,j}$, while table $Q_{i,j}^{bp}$ holds the values of the partition function of the sub-sequences for a base pair or 0 when base pairing is absent.

The following calculations are used to compute the values of the partition functions and inserted as elements of tables $Q_{i,j}$ and $Q_{i,j}^{bp}$.

$$Q_{i,j} = Q_{i,j-1} + \sum_{i \leq k < (j-1)} Q_{i,k-1} \cdot Q_{k,j}^{bp}$$

Listing 1. Code of the McCaskill partition function computation.

```

if (N>5 && l>=0 && l<=5)
  for (i=N-1; i>=0; i--)
    for (j=i+1; j<N; j++){
      Q[i][j] = Q[i][j-1];
      for (k=0; k<j-i-1; k++){
        Qbp[k+i][j] = Q[k+i+1][j-1] *
                      ERT * paired(k+i, j-1);
        Q[i][j] += Q[i][k+i] * Qbp[k+i][j];
      }
    }
    
```

$$Q_{i,j}^{bp} = \begin{cases} Q_{i+1,j-1} \cdot \exp(-E_{bp}/RT) & \text{if } S_i, S_j \text{ can form} \\ & \text{base pair} \\ 0 & \text{otherwise} \end{cases}$$

Listing 1 presents the code implementation computing $Q_{i,j}$ and $Q_{i,j}^{bp}$ filled with random double values (data in arrays do not affect the speed of the code). ERT is equal to $\exp(-E_{bp}/RT)$. To simplify target tiled code generation, we replaced k with $k+i$ and the innermost loop boundaries from 0 to $j-i-1$.

Many algorithms in bioinformatics are within dynamic programming (DP). Programming loop nests implementing those algorithms can be represented within the polyhedral model. That model is used in many optimization compilers, which automatically generate efficient parallel tiled code. However, the code implementing McCaskill’s algorithm exposes non-uniform dependences that make it difficult effective parallelization and tiling of that code.

A polyhedral optimizer generally improves code locality by means of loop tiling, which groups loop statement instances within smaller blocks (tiles). This allows for reuse provided that the block fits in cache. In parallel tiled code, tiles are enumerated as indivisible macro statements. This increases the granularity of parallel code that often improves the performance of that code executed in parallel systems with shared memory.

Dynamic programming codes expose non-uniform dependences, which limit applying commonly known optimization techniques such as permutation, diamond tiling [3], or index

set splitting [4] very well trained, for example, on stencils.

II. OPTIMIZING COMPILERS USED FOR EXPERIMENTS

Modern automatic optimizing compilers, for example, PLuTo [5], demonstrate the success of using the polyhedral model. PLuTo extracts and applies affine functions to parallelize and tile serial code. Target parallel tiled code demonstrates good efficiency on modern multi-core computers with shared memory in particular for stencils.

For a given loop nest statement, compilers based on affine transformations use the relation $[I] \rightarrow [t = C * I + c]$, where I is the loop statement iteration vector; t is the time assigned to execute iteration I ; $C * I + c$ represents the affine function. When two statement instances get the same execution time, they can be run in parallel. To extract the unknown matrix C and unknown vector c , firstly, for each loop nest statement, time-partition constraints are formed by applying extracted dependences. Then the time-partition constraints are resolved for elements of matrix C and elements of vector c .

The PLuTo engine is used in other compilers, for example, in Apollo[6], PPCG [7], PTile[8], and Autogen framework [9] as well as in commercial IBM-XL and R-STREAM from the Reservoir Lab [10].

TRACO is based on the slicing framework introduced in paper [11]. It calculates the transitive closure of a dependence graph, which is used to fulfill corrections of original rectangular tiles. The goal of the correction is to eliminate all cycles among target tiles. This allows us to enumerate target tiles in lexicographic order.

After tile correction, the inter-tile dependence graph does not contain any cycle and any technique of loop nest parallelization can be used to generate parallel code, details are presented in paper [12]. TRACO uses the commonly known wave-fronting technique for tiled code parallelization. For its implementation, it applies the ISL library.

PLuTo and TRACO have some limitations. PLuTo may not find the number of linearly independent solutions to time partitions constraints that is equal to the number of the loops surrounding a loop nest statement. This reduces the dimension of target tiles and as a consequence target code locality may be low. For example, it is not able to tile each loop nest for the Nussinov and Knuth algorithms [13], i.e., instead of 3D tiles it generates only 2D tiles. TRACO, in general, may generate irregular tiles that reduce code locality and worsen multiple thread work balance [14].

To generate regular code and increase tile dimension, DAPT implements space-time tiling. First DAPT generates space tiles according to the technique presented in paper [15]. Then it splits each space tile into multiple time slices. Each time slice is represented with a number of time partitions found by means of the ISL scheduler. The number of time partitions within the time slice is defined by the user. As a result, the tile dimension is increased by one. Target code enumerates smaller tiles (time slices) within each space tile. This increases code locality due to increasing the probability of catching all the

data associated with each smaller tile in cache provided that the number of time partitions forming the time slice is chosen properly.

However, each of the mentioned above automatic source-to-source compilers is able to generate tiled code for the McCaskill algorithm and we conducted a comparison analysis of the performance of codes generated by them on two modern multi-core machines.

III. EXPERIMENTAL STUDY

In this section, we present the results of an experimental study with PLuTo, TRACO, and DAPT codes implementing the McCaskill partition function computation. All target parallel tiled codes were compiled using the Intel C++ Compiler (icc) and GNU C++ Compiler (g++) with the -O3 flag of optimization.

To carry out experiments, we used two multi-processor machines: an 2x Intel Xeon Platinum 9242 CPU (2.30GHz, 2x96 threads, 71,5 MB Cache, compiler icc 21.3.0, 2019), and an AMD Epyc 7542 (2.35 GHz, 32 cores, 64 threads, 128MB Cache, compiler g++ 9.3.0, 2019).

The code generated with DAPT is presented in Listing 2, while the codes generated with PLuTo and TRACO can be found at https://github.com/markpal/NPDP_Bench/blob/main/mcc/mcc_dapt.cpp, they are too long to be inserted in this paper.

It is worth noting that tiles generated with TRACO are irregular, they can be fixed or parametric (the size of such tiles is unbounded). PLuTo generates regular fixed tiles except from boundary ones.

Space-time tiling implemented in DAPT generates regular tiles and the tile dimension is one more than that of tiles generated with PLuTo.

In all examined compilers, for parallelism representation, the OpenMP standard is used. For different sizes examined by us, by means of experiments, we discovered that the best tile size for the TRACO target code is 1x128x16. This means that the outermost loop in the loop nest should not be tiled. For the target code generated with PLuTo, the best tile size is 16x16x16. For the DAPT code, the optimal size is 16x16x16 for space slices and the size of the time slice (the number of time partitions within the space tile) is 2.

The McCaskill code can be tiled by all the compilers used for us for experiments. However, only TRACO and DAPT allow us to generate parallel tiled code. The serial code generated with PLuTo is very cache-efficient, but PLuTo is unable to extract any affine schedule allowing for parallelism of target code. TRACO generates target code applying the transitive closure of the dependence graph for the McCaskill loop nest, then it builds a relation representing inter-tile dependences. Finally, using that relation, it applies the ISL scheduler to extract a valid tile schedule, which is used to generate parallel tiled code. DAPT applies the wave front technique to generate target parallel tiled code.

Tables 1 and 2 hold execution times (in seconds) for the PLuTo, TRACO, and DAPT codes for various RNA sequence

Listing 2. Parallel tiled loop nests of the McCaskill algorithm generated with DAPT.

```

1  if ( l >= 0 && l <= 5 && N >= 6 ) {
2  for(w0 = -1; w0 <= (N - 1) / 8; w0 += 1) {
3  #pragma omp parallel for
4  for(h0 = max(w0 - (N+7) / 8 + 1, -((N+5) / 8)); h0 <= min(0, w0); h0 += 1) {
5  for(i0 = max(max(-N+2, -8*w0 + 8*h0-6), 8*h0); i0 <= min(0, 8*h0+7); i0++) {
6  for(i1 = max(8*w0 - 8*h0, -i0+1); i1 <= min(N-1, 8*w0 - 8*h0 + 7); i1++) {
7  Q[-i0][i1] = Q[-i0][i1 - 1];
8  for (i3 = 0; i3 < -1 + i0 + i1; i3 += 1) {
9  Qbp[-i0+i3][i1] = ((Q[-i0+i3+1][i1-1] * (ERT)) * paired((-i0+i3), (i1-1)));
10 Q[-i0][i1] += (Q[-i0][-i0 + i3] * Qbp[-i0 + i3][i1]);
11 }}}}

```

TABLE I
EXECUTION TIME OF THE PARALLEL TILED CODES FOR AN INTEL XEON PLATINUM 9242 USING 192 HARDWARE THREADS.

N	Serial	PLuTo	TRACO	Dapt
1000	0,61	0,51	0,14	0,07
2000	8,81	4,73	0,87	0,62
3000	38,63	21,73	3,43	2,22
4000	106,17	58,96	8,58	5,06
5000	227,41	116,75	17,73	9,88
6000	420,26	206,48	29,43	17,45
7000	721,88	333,84	47,61	28,16
8000	1157,84	503,13	69,81	46,41
9000	2575,45	713,69	98,67	71,33
10000	3676,3	1005,44	135,01	105,86

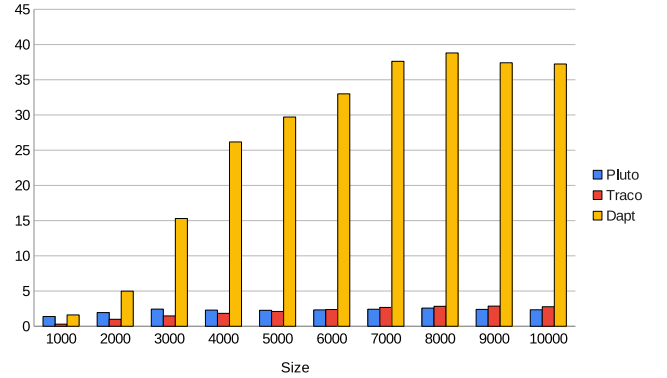


Fig. 1. Speedups of the parallel tiled codes generated by applying TRACO, PLuTo, and Dapt for an Intel Xeon Platinum 9242 and RNA sequence sizes from 1000 to 10000.

lengths on an Intel Xeon Platinum 9242 (192 hardware threads) and an AMD Epyc 7542 (64 hardware threads), respectively. Figures 1 and 3 show the speed-up (a ratio of T_1 over T_n , where T_1 and T_n are execution times for one and n threads used for code running, respectively) of parallel tiled programs for RNA sequence sizes from 1000 to 10000 for an Intel Xeon Platinum 9242 (192 hardware threads) and an AMD Epyc 7542 (64 hardware threads), respectively. Figures 2 and 4 show how target code speed-up depends on the number of threads for an Intel Xeon Platinum 9242 (192 hardware threads) and an AMD Epyc 7542 (64 hardware threads), respectively, for $N = 5000$ (roughly the size of the longest human mRNA).

Analyzing the presented results of experiments, we may state that the DAPT code overcomes considerable those of PLuTo and TRACO ones. The TRACO code overcomes that of PLuTo for eight and more threads. The worse efficiency of the TRACO code for a few thread numbers is caused with the irregularity of the target code (see the previous section).

IV. CONCLUSION

We presented the results of a comparative performance analysis of three tiled codes generated with optimizing compilers PLuTo, TRACO, and DAPT for the McCaskill partition function calculation. The best performance demonstrates the DAPT code due to the fact that it applies space-time tiling

TABLE II
EXECUTIONS TIME OF THE PARALLEL TILED CODES FOR AN AMD EPYC 7542 USING 64 HARDWARE THREADS.

N	Serial	PLuTo	TRACO	Dapt
1000	0,87	0,63	2,89	0,54
2000	10,22	5,28	10,33	2,05
3000	46,19	18,99	31,46	3,02
4000	129,29	56,33	70,56	4,94
5000	289,41	127,89	137,64	9,74
6000	572,61	246,36	241,06	17,35
7000	999,91	413,36	373,89	26,58
8000	1567,43	607,81	553,21	40,39
9000	2231,09	932,75	779,01	59,63
10000	3043,21	1299,29	1096,88	81,72

allowing us to increase the tile dimensionality by one in comparison with that of PLUTO. That makes all tiles to be regular and of fixed size. A proper choice of a tile size allows us to hold all the data associated with each tile in cache that increases code locality.

REFERENCES

- [1] M. Raden, S. M. Ali, O. S. Alkhnbashi, A. Busch, F. Costa, J. A. Davis, F. Eggenhofer, R. Gelhausen, J. Georg, S. Heyne, M. Hiller, K. Kundu,

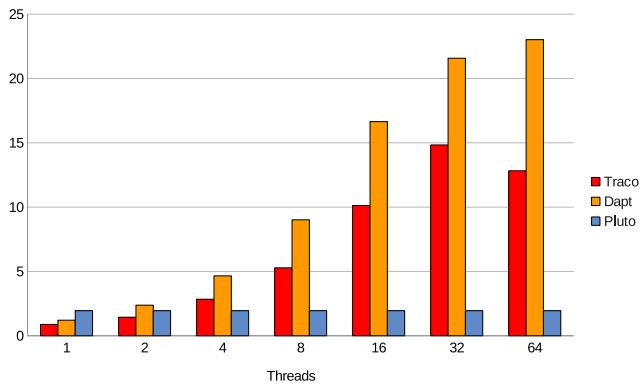


Fig. 4. Speedups of the parallel tiled codes generated by applying TRACO, PLuTo, and Dapt for an AMD Epyc 7542 using various number of hardware threads

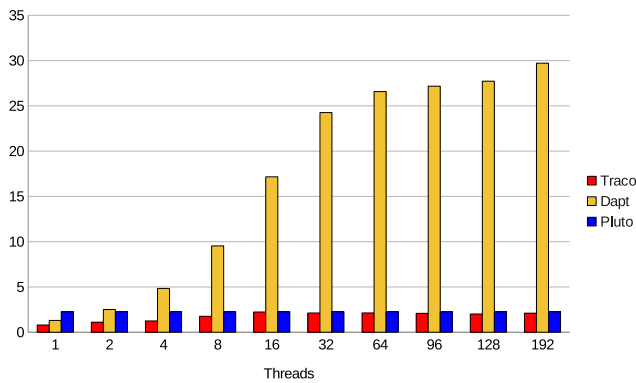


Fig. 2. Speedups of the parallel tiled codes generated by applying TRACO, PLuTo, and Dapt for an Intel Xeon Platinum 9242 using various number of hardware threads.

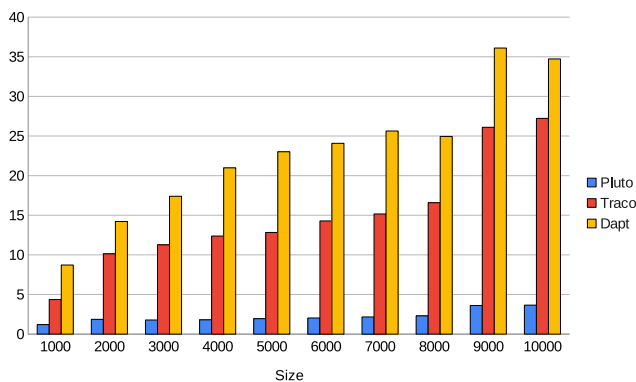


Fig. 3. Speedups of the parallel tiled codes generated by applying TRACO, PLuTo, and Dapt for an AMD Epyc 7542 and RNA sequence sizes from 1000 to 10000.

- R. Kleinkauf, S. C. Lott, M. M. Mohamed, A. Mattheis, M. Miladi, A. S. Richter, S. Will, J. Wolff, P. R. Wright, and R. Backofen, "Freiburg RNA tools: a central online resource for RNA-focused research and teaching," *Nucleic Acids Research*, vol. 46, no. W1, pp. W25–W29, 2018. doi: 10.1093/nar/gky329
- [2] M. Raden, S. M. Ali, O. S. Alkhnabshi, A. Busch, F. Costa, J. A. Davis, F. Eggenhofer, R. Gelhausen, J. Georg, S. Heyne *et al.*, "Freiburg rna tools: a central online resource for rna-focused research and teaching," *Nucleic acids research*, vol. 46, no. W1, pp. W25–W29, 2018.
- [3] U. Bondhugula, V. Bandishti, and I. Pananilath, "Diamond tiling: Tiling techniques to maximize parallelism for stencil computations," *IEEE Transactions on Parallel and Distributed Systems*, vol. 28, no. 5, pp. 1285–1298, May 2017. doi: 10.1109/tpds.2016.2615094
- [4] U. Bondhugula, A. Acharya, and A. Cohen, "The pluto+ algorithm: A practical approach for parallelization and locality optimization of affine loop nests," *ACM Trans. Program. Lang. Syst.*, vol. 38, no. 3, pp. 12:1–12:32, Apr. 2016. doi: 10.1145/2896389
- [5] U. Bondhugula *et al.*, "A practical automatic polyhedral parallelizer and locality optimizer," *SIGPLAN Not.*, vol. 43, no. 6, pp. 101–113, Jun. 2008. doi: 10.1145/1379022.1375595 [Http://pluto-compiler.sourceforge.net/](http://pluto-compiler.sourceforge.net/).
- [6] J. M. M. Caamaño, A. Sukumaran-Rajam, A. Baloian, M. Selva, and P. Clauss, "Apollo: Automatic speculative polyhedral loop optimizer," in *IMPACT 2017-7th International Workshop on Polyhedral Compilation Techniques*, 2017, p. 8.
- [7] S. Verdoolaege, J. Carlos Juega, A. Cohen, J. Ignacio Gómez, C. Tenllado, and F. Catthoor, "Polyhedral parallel code generation for cuda," *ACM Trans. Archit. Code Optim.*, vol. 9, no. 4, jan 2013. doi: 10.1145/2400682.2400713. [Online]. Available: <https://doi.org/10.1145/2400682.2400713>
- [8] M. M. Baskaran, A. Hartono, S. Tavarageri, T. Henretty, J. Ramanujam, and P. Sadayappan, "Parameterized tiling revisited," in *Proceedings of the 8th annual IEEE/ACM international symposium on Code generation and optimization*, ser. CGO '10. New York, NY, USA: ACM, 2010. ISBN 978-1-60558-635-9 pp. 200–209.
- [9] R. Chowdhury, P. Ganapathi, J. J. Tithi, C. Bachmeier, B. C. Kuzmaul, C. E. Leiserson, A. Solar-Lezama, and Y. Tang, "Autogen: Automatic discovery of cache-oblivious parallel recursive algorithms for solving dynamic programs," *ACM SIGPLAN Notices*, vol. 51, no. 8, pp. 1–12, 2016.
- [10] U. Bondhugula *et al.*, "A practical automatic polyhedral parallelizer and locality optimizer," *SIGPLAN Not.*, vol. 43, no. 6, pp. 101–113, Jun. 2008. [Online]. Available: <http://pluto-compiler.sourceforge.net>
- [11] W. Pugh and D. Wonnacott, "An exact method for analysis of value-based array data dependences," in *Sixth Annual Workshop on Programming Languages and Compilers for Parallel Computing*. Springer-Verlag, 1993.
- [12] W. Bielecki and M. Palkowski, "Tiling of arbitrarily nested loops by means of the transitive closure of dependence graphs," *International Journal of Applied Mathematics and Computer Science (AMCS)*, vol. Vol. 26, no. 4, pp. 919–939, December 2016. doi: 10.1515/amcs-2016-0065
- [13] W. Bielecki and M. Palkowski, "Space-time loop tiling for dynamic programming codes," *Electronics*, vol. 10, no. 18, p. 2233, 2021.
- [14] M. Palkowski and W. Bielecki, "Parallel cache-efficient code for computing the McCaskill partition functions," vol. 18, pp. 207–210, 2019. doi: 10.15439/2019F8
- [15] W. Bielecki and M. Poliwoda, "Automatic parallel tiled code generation based on dependence approximation," in *International Conference on Parallel Computing Technologies*. Springer, 2021, pp. 260–275.

Influence of loop transformations on performance and energy consumption of the multithreaded WZ factorization

Beata Bylina, Jarosław Bylina, Monika Piekarz
Institute of Computer Science, Marie Curie-Skłodowska University
Pl. M. Curie-Skłodowskiej 5
Lublin, 20-031, Poland
Email: {beata.bylina, jaroslaw.bylina, monika.piekarz}@umcs.pl

Abstract—High-level loop transformations are a key instrument to effectively exploit the resource in modern architectures. Energy consumption on multi-core architectures is one of the major issues connected with high-performance computing. We examine the impact of four loop transformation strategies on performance and energy consumption. The investigated strategies include: loop fission, loop interchange (permutation), strip-mining, and loop tiling. Additionally, a column-wise and row-wise store formats for dense matrices are considered. Parallelization and vectorization are implemented using OpenMP directives. As a test, the WZ factorization algorithm is used. The comparison of selected strategies of the loop transformation is done for Intel architecture, namely Cascade Lake. It has been shown that for WZ factorization, which is an example of an application in which we can use the loop transformation, optimization towards high-performance can also be an effective strategy for improving energy efficiency. Our results show also that block size selection in loop tiling has a significant impact on energy consumption.

Keywords: energy saving, energy consumption, RAPL, WZ factorization, multicore architecture

I. INTRODUCTION

WITH the growing demand for high-performance computing, new architectures have emerged which unfortunately consumes more and more energy. Reducing energy consumption in these architectures is one of the major challenges. The current research trends based on performance studies [12], [13], [16] and comparisons are to develop hardware and software to achieve the best performance and energy compromise. One of the aspects of creating energy-aware software is the optimization of implementation of complex numerical algorithms. Such complex algorithms include loops, in particular nested ones. Nested loops are an important structure bearing a great deal of the parallelism and vectorization possibilities. However, to parallelize them efficiently, the programmer has to make some decisions about applying various transformations. An example of such loops is matrix algorithms, like matrix multiplication or different kinds of factorizations, widely investigated in the literature [3], [1].

In the work [2], we studied loop transformations for nested loops on multicore architectures on the example of a factorization similar to the LU factorization, namely, the WZ factorization [19]. The WZ factorization has some nontrivial

data dependencies and the compiler is not able to efficiently optimize the algorithm. We have chosen the following four: loop fission, loop interchange (loop permutation), strip-mining, and loop tiling.

In this article, we investigate the impact of these four loop transformations on performance and energy consumption for the WZ factorization on multicore architecture. We describe in detail two block-related transformations (strip-mining and loop tiling). We are making theoretical and experimental considerations about the size of the blocks. Additionally, we consider column-wise and row-wise storage formats for dense matrices. The OpenMP standard is used for parallelization and vectorization of the code. The Intel RAPL (Running Average Power Limit) [7], [9] interface is used as a source of information on energy consumption.

The main contributions of this article are the following:

- results of the tests from the evaluation of the execution time and energy consumption for four loop transformations of the WZ factorization for various data sizes on Intel architecture — namely, Cascade Lake;
- conclusions on the impact of the four loop transformations on the execution time and energy consumption;
- analysis of the correlation between performance and energy consumption for four loop transformations.

This paper is organized as follows. Section II presents a few related works on loop transformations and energy consumption/performance analysis for modern computer architectures and systems. Section III discusses the four loop transformations applied here for the WZ factorization and studies block size for tiled transformation. In Section IV, we concentrate on the details of conducting tests and on the discussion and explanation of the results. In Section V, we present the conclusions of the numerical experiments and further research directions.

II. RELATED WORKS

When high-performance computing is considered, energy consumption is one of the most important challenges. These challenges are analyzed on many levels. In particular, the works of [11], [13], [16] dealt with the topic of energy

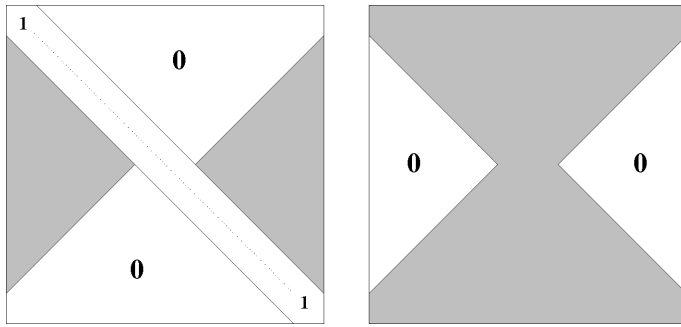


Fig. 1: The output of the WZ factorization — forms of the matrices \mathbf{W} (left) and \mathbf{Z} (right).

consumption in the context of using the OpenMP standard for multi-core computers with shared memory.

The key to software optimization in terms of the performance for algorithms that include nested loops is the right choice of the appropriate loop transformations. Loop transformations are a research topic for various automatic optimization techniques [8], [10], [14], [15] as well as for manual conversion of application code [5], [17], [18] so as to obtain the best possible performance on modern multi-core architectures.

In many numerical algorithms where dependencies between data are very complicated, even such tools as efficient optimizing compilers are not able to transform the code to use the potential of modern processors. The authors of [3], [1], present algorithms for solving systems of equations, trying to improve their performance, in particular in parallel. Improvement in performance was obtained by appropriate transformation of the underlying algorithm using looping tiling and appropriate data structures. There are currently not too many studies in the literature about both efficiency and energy consumption in the context of loop transformations. In our work, we study a numerical algorithm (the WZ factorization), in which, loops are transformed. This algorithm concerns numerical linear algebra, in particular solving systems of equations on multi-core architectures using OpenMP in the context of performance and energy consumption.

III. WZ FACTORIZATION

The WZ factorization (Figure 1) was introduced in [4]. It was a new method for parallel solving of systems of linear equations on computers containing many data processing units and it was an alternative to the well-known LU factorization.

The WZ factorization is based on the decomposition of the square matrix \mathbf{A} into two matrices: \mathbf{W} i \mathbf{Z} . All the matrices which we consider are dense ones. They are stored as one-dimensional arrays in one of the two formats: column-wise or row-wise.

The basic algorithm for the WZ factorization for an even size of the matrix (we only consider even sizes — without loss of generality) is shown in Figure 2.

The loop transformation consists in replacing itself with an equivalent loop containing the structured block. Some

```

for(k = 0; k < n/2-1; k++) {
    p = n-k-1;
    akk = a[k][k];    akp = a[k][p];
    apk = a[p][k];    app = a[p][p];
    detinv = 1 / (apk*akp - akk*app);
    for(i = k+1; i < p; i++) {
        w[i][k] = (apk*a[i][p] - app*a[i][k])
                * detinv;
        w[i][p] = (akp*a[i][k] - akk*a[i][p])
                * detinv;
        for(j = k+1; j < p; j++)
            a[i][j] = a[i][j]
                    - w[i][k]*a[k][j]
                    - w[i][p]*a[p][j];
    }
}

```

Fig. 2: The basic algorithm for the WZ factorization — pseudocode.

```

for(k = 0; k < n/2-1; k++) {
    .
    .
    for(i = k+1; i < p; i++) {
        w[i][k] = (apk*a[i][p] - app*a[i][k])
                * detinv;
        w[i][p] = (akp*a[i][k] - akk*a[i][p])
                * detinv;
    }
    for(i = k+1; i < p; i++)
        for(j = k+1; j < p; j++)
            a[i][j] = a[i][j]
                    - w[i][k]*a[k][j]
                    - w[i][p]*a[p][j];
}

```

Fig. 3: The algorithm after the fission of the i -loop — pseudocode. This algorithm matches the row-wise layout.

well-known transformations considered are: loop fission, loop interchange (permutation), strip-mining, tiling.

A. Loop fission and permutation

Loop fission (also called loop distribution) breaks a loop into multiple loops over the same index range but each taking only a part of the loop's body. Its purpose is to achieve better utilization of locality of reference — isolate parallelizable loops create independent loops, hence creating separate tasks. Its result is shown in Figure 3.

In the fission algorithm (Figure 3) it is possible to use the loop interchange (for the j -loop and the second i -loop). The loop interchange transformation switches the order of loops' nesting (consists in replacing the internal loop with the external one). The purpose of such a transformation is to improve data locality or increase parallelism and vectorization. The algorithm from Figure 3 is denoted as `fission-ij`. Its result is shown in Figure 4.

```

for(k = 0; k < n/2-1; k++) {
    .
    .
    for(i = k+1; i < p; i++) {
        w[i][k] = (apk*a[i][p] - app*a[i][k])
                * detinv;
        w[i][p] = (akp*a[i][k] - akk*a[i][p])
                * detinv;
    }
    for(j = k+1; j < p; j++)
        for(i = k+1; i < p; i++)
            a[i][j] = a[i][j]
                    - w[i][k]*a[k][j]
                    - w[i][p]*a[p][j];
}

```

Fig. 4: The algorithm after the fission of the i -loop with permutation loop — pseudocode. This algorithm matches the row-wise layout.

B. Strip-mining and loop tiling

Access to the main memory in our algorithm takes a lot of time. It is a well-known fact that the cost of accessing the memory is much higher than the cost of computations. We can even figure the number of memory reads and writes (C_M) and compare it to the number of floating-point operations (C_F). After some simple calculations we obtain:

$$C_M = \frac{7}{6}n^3 + O(n^2),$$

$$C_F = \frac{2}{3}n^3 + O(n^2).$$

Thus, the ratio of memory access to computations is:

$$\frac{C_M}{C_F} \approx \frac{7}{4}.$$

This means that we need a lot of memory access to perform our algorithm — almost two memory accesses for one floating-point operation. So, it is the main obstacle to utilizing the computing power of modern processors fully. A manner to solve this problem is to use the cache memory (which is much faster than the main memory) efficiently. Of course, the size of the cache memory is too small to house all the data needed in the algorithm.

Strip-mining is a loop transformation that consists in replacing one loop with two nested loops. One of them (inner) is appropriate for vectorization (it is quite short and with a unit stride), and another (outer) is longer and its step is equal to the full number of iterations in the inner one. The transformation pays only when the original loop is rather long.

A loop in the process of strip-mining is divided into two loops, where the inner one has $BLOCK_SIZE$ iterations and the outer one has $n/BLOCK_SIZE$ iterations (n is the number of iterations in the original loop). The strip-mining alone can have some positive impact on the performance (by easing the automatic vectorization process).

One of the widely used techniques which allow for improving performance is loop tiling which consists in connecting

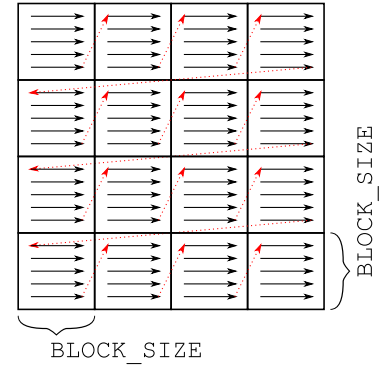


Fig. 5: Computing sequence after loop tiling (black: computations in blocks; red: order between blocks).

strip-mining with loop interchange. The main aim of this technique is to reduce the number of reads from and writes to the main memory by improving the spatial and temporal data locality, and hence by better utilizing the hierarchy of the memory — especially the cache memory by lessening the rate of cache misses. It is also useful for vectorization — both automatic and explicit — and for parallelization.

After such a transformation, the order of the computations changes (see Figure 5), although without the change of the result.

In such a process we improve the temporal and spatial locality of the data. By dividing the data into pieces of $BLOCK_SIZE$, we cause them to fit in cache memory (we mean level 1 cache here) and stay there as long as needed to conduct current computations. This minimizes the frequency of cache memory swaps. Too big $BLOCK_SIZE$ and the data would not fit into the cache, too small $BLOCK_SIZE$ and the swapping frequency rises. Moreover, to facilitate the vectorization, we wanted to make $BLOCK_SIZE$ a multiple of the SIMD register. Unfortunately, in most of the iterations of both the i -loop and j -loop, the first and the last iteration is chopped. For this reason, we broadened the first and the last iteration to full $BLOCK_SIZE$. It does not change the results of the algorithm (additional operations are beyond the non-zero elements of the resulting matrices), although, it forces the machine to make some more computations (in Figure 6, the red color denotes the redundant computations) — the bigger $BLOCK_SIZE$ the more additional computations are needed. The advantage is the possibility of vectorizing evenly all the iterations.

To achieve this, we make every outer loop start with a nearest full multiple of $BLOCK_SIZE$ (rounded down) and we make every inner loop iterate through a whole $BLOCK_SIZE$. Moreover, all the matrices were allocated with the memory alignment suitable for the used architecture.

Figure 7 shows the original algorithm after parallelization, with strip-mining of the j -loop (the full loop tiling is impossible due to the infeasibility of the loop interchange). Figure 8 presents the fission algorithm with full loop tiling. The function $RDTTNM()$ (stands for *round down to the nearest*

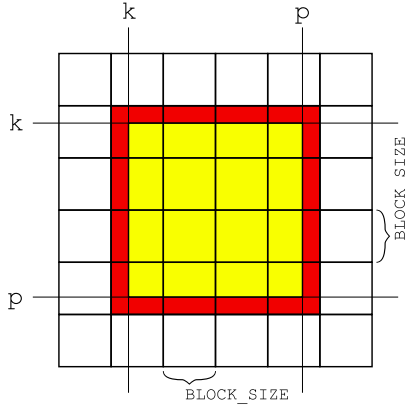


Fig. 6: Computations in the k th step of the algorithm after loop tiling (yellow: necessary computations; red: needless computations).

```

for(k = 0; k < n/2-1; k++) {
    . . .
    for(i = k+1; i < p; i++) {
        w[i][k] = (apk*a[i][p] - app*a[i][k])
            * detinv;
        w[i][p] = (akp*a[i][k] - akk*a[i][p])
            * detinv;
        start = RDTNM(k+1, BLOCK_SIZE);
        for(jj = start; jj < p;
            jj += BLOCK_SIZE) {
            __assume(jj % BLOCK_SIZE == 0);
            for(j = jj; j < jj+BLOCK_SIZE;
                ++j)
                a[i][j] = a[i][j]
                    - w[i][k]*a[k][j]
                    - w[i][p]*a[p][j];
        }
    }
}

```

Fig. 7: Strip-mining in the basic algorithm — pseudocode.

multiple) can be defined as a macro:

```
#define RDTNM(a, r) (((a)/(r))* (r))
```

In Figures 7 and 8, we use the compiler clause `__assume` which tells the compiler that a given condition is fulfilled — here, we declare that `ii` and `jj` are multiples of the `BLOCK_SIZE` which facilitates the vectorization.

Table I shows the computation overhead for the algorithm after loop tiling (red in Figure 6) — as a function of the size n of the matrix and of the `BLOCK_SIZE` (b).

Figure 9 shows the dependencies (black arrows) between the input data (green and blue dots) and the results (red dots) of the innermost loop in the k th step of the WZ factorization. We can see that — with no regard to the matrix memory layout (even for more complex layouts [6]) — the results of the computations depend on the data from various areas of the memory, so the loop tiling can give very limited performance improvements.

Here, the OpenMP standard is used for parallelization and vectorization code for all loop transformations. The more outer `i`-loop or `ii`-loop is parallelized with the `pragma parallel`

```

for(k = 0; k < n/2-1; k++) {
    . . .
    for(i = k+1; i < p; i++) {
        w[i][k] = (apk*a[i][p] - app*a[i][k])
            * detinv;
        w[i][p] = (akp*a[i][k] - akk*a[i][p])
            * detinv;
    }
    start = RDTNM(k+1, BLOCK_SIZE);
    for(ii = start; ii < p;
        ii += BLOCK_SIZE) {
        for(jj = start; jj < p;
            jj += BLOCK_SIZE) {
            __assume(ii % BLOCK_SIZE == 0);
            for(i = ii; i < ii+BLOCK_SIZE;
                ++i) {
                __assume(
                    jj % BLOCK_SIZE == 0);
                for(j = jj; j < jj+BLOCK_SIZE;
                    ++j)
                    a[i][j] =
                        a[i][j]
                        - w[i][k]*a[k][j]
                        - w[i][p]*a[p][j];
            }
        }
    }
}

```

Fig. 8: Loop tiling in the fission algorithm — pseudocode.

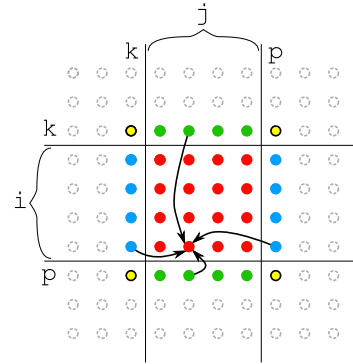


Fig. 9: The arrows show the data dependencies for updating an element of the array a in the k th step. Grey, green and yellow elements have their final values in the k th step; yellow ones are needed to compute `detinv` in this step; blue ones are elements of the array w computed in the middle loop of this step; red ones are updated in the innermost loop with the use of two blue elements and two green elements each.

loop and the most inner `j`-loop is vectorized with the `pragma simd` — details in [2].

IV. NUMERICAL EXPERIMENT – METHODOLOGY AND RESULTS ANALYSIS

A. Methodology

We test two types of versions of the WZ factorization algorithm:

- the basic algorithm presented in Figure 2 for the matrix stored in memory in two formats: a row-wise (basic-row) and column-wise (basic-col);

TABLE I: The ratio of the excess computations in the algorithm after loop tiling to the computations in the basic algorithm (n is the size of the matrix; b is the BLOCK_SIZE).

	$b = 8$	16	32	64	128	256	512
$n = 1024$	2.06%	4.44%	9.28%	19.24%	40.33%	87.21%	199.71%
2048	1.03%	2.21%	4.59%	9.42%	19.38%	40.48%	87.35%
3072	0.68%	1.47%	3.05%	6.24%	12.75%	26.29%	55.46%
4096	0.51%	1.10%	2.28%	4.66%	9.50%	19.46%	40.55%
5120	0.41%	0.88%	1.82%	3.72%	7.57%	15.44%	31.94%
6144	0.34%	0.73%	1.52%	3.10%	6.29%	12.80%	26.34%
7168	0.29%	0.63%	1.30%	2.65%	5.38%	10.93%	22.41%
8192	0.26%	0.55%	1.14%	2.32%	4.70%	9.53%	19.49%
9216	0.23%	0.49%	1.01%	2.06%	4.17%	8.46%	17.25%
10240	0.21%	0.44%	0.91%	1.85%	3.75%	7.60%	15.47%
11264	0.19%	0.40%	0.83%	1.68%	3.41%	6.89%	14.02%
12288	0.17%	0.37%	0.76%	1.54%	3.12%	6.31%	12.82%
13312	0.16%	0.34%	0.70%	1.42%	2.88%	5.82%	11.81%
14336	0.15%	0.31%	0.65%	1.32%	2.67%	5.40%	10.95%
15360	0.14%	0.29%	0.61%	1.23%	2.49%	5.04%	10.20%
16384	0.13%	0.27%	0.57%	1.16%	2.34%	4.72%	9.55%
32768	0.06%	0.14%	0.28%	0.58%	1.17%	2.35%	4.73%

- the algorithms with the fission loop transformation and loop interchange, that is: `fission-row-ij` (Figure 3) and `fission-row-ji` (Figure 4) for the row-wise layout and `fission-col-ij`, `fission-col-ji` for the column-wise layout;

and two types of versions of WZ factorization block algorithms:

- the strip-mining algorithms: `basic-row-sm-b`, `basic-col-sm-b`;
- the loop tiling algorithms: `fission-row-ij-lt-b`, `basic-row-ji-lt-b`, `fission-col-ij-lt-b`, `fission-col-ji-lt-b`.

In the notation `sm` is short for strip-mining, `lt` for loop tiling, and b is the BLOCK_SIZE. The following block sizes are checked: 8, 16, 32, 64, 128, 256, 512.

All versions have been implemented in C++ with vectorization and parallelization.

For testing, we used a double-precision square matrix of random values. The size of the smallest matrix is R (rows times columns). All sizes are shown in Table II.

TABLE II: Characteristics of the test data sizes.

Data size	n	Number of cells ($n \times n$)	[GB]
R	8192	67108864	0.5
2.25R	12288	150994944	1.125
4R	16384	268435456	2
16R	32768	1073741824	8

The performance and energy consumption tests of the proposed versions of the WZ algorithm were carried out on the following computing platform equipped with a modern multi-core processor with the following parameters and software:

- processor: Intel(R) Xeon(R) Gold 5218R (2.10 GHz; HT; 2×20 cores);
- operating system: CentOS 7.5 with Linux kernel 3.10.0;

- compiler: Intel ICC 14.0.2 with compiler options
`-qopenmp -O3 -ipo -no-prec-div`
`-fp-model fast=2`

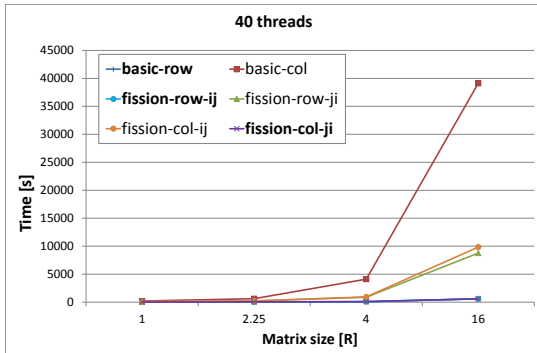
To analyze the impact of all versions of the algorithm on energy consumption, we used measurements from the RAPL (Running Average Power Limit) interface. We used RAPL because the article [9] has shown that it gives the correct measurement results.

B. Execution Time. Matrix layout and loop interchange

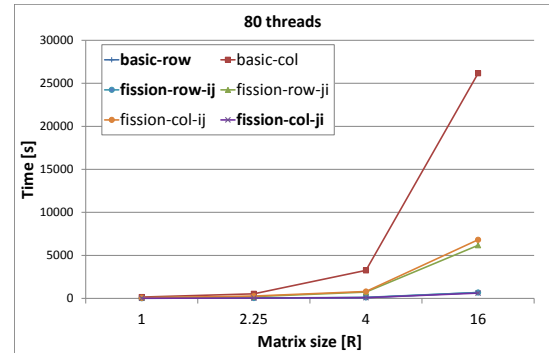
First, we measure the execution time of the following versions of WZ factorization algorithm: the basic algorithm and the loop fission algorithm for the matrix stored in memory in two formats: a row-wise and column-wise manner and loop interchanged the order of the i -loop and j -loop. Therefore, we test 6 versions of the algorithm:

- `basic-row`
- `basic-col`
- `fission-row-ij`
- `fission-row-ji`
- `fission-col-ij`
- `fission-col-ji`

We test all these versions on 40 threads as well as on 80 threads with running HT. The results are presented in Figure 10. Both graphs on the vertical axis have execution time and on the horizontal axis the data size. Also, both of them show lines with points indicating the time of the algorithm for different data sizes. The graph in Figure 10a shows the runtime of the algorithm running on 40 threads, whereas the graph in Figure 10b shows the runtime of 80 threads, respectively. In both graphs we see 6 lines, each of them concerns one version of the algorithm according to the legend. Also in both cases, the lines for the `basic-row`, `fission-row-ij`, and `fission-col-ji` algorithms almost overlap. As we can see from the graphs, the above-mentioned versions also have the shortest runtime. Table III shows the speedup we get



(a) 40 threads



(b) 80 threads

Fig. 10: Execution time of versions of WZ factorization algorithm for different data sizes.

on 80 threads compared to 40 threads for each version of the tested WZ algorithm, which is defined as follows:

$$S = \frac{T_{40th}}{T_{80th}}$$

where parameter T_{40th} denotes the execution time of the algorithm run on 40 threads and T_{80th} denotes the execution time of the algorithm run on 80 threads.

TABLE III: Relative speedup of versions of WZ algorithm operating on 40 and 80 threads (T_{40th}/T_{80th}).

	R	2.25R	4R	16R
basic-row	0.91	0.95	0.88	0.85
basic-col	1.21	1.15	1.26	1.49
fission-row-ij	0.93	0.81	0.70	0.85
fission-row-ji	1.11	1.03	1.21	1.41
fission-col-ij	1.22	0.88	1.15	1.44
fission-col-ji	0.96	0.89	0.90	0.96

Based on Table III, we can observe that, regardless of the size of data, the acceleration of the operating time between the operation on 40 threads and on 80 threads is of a similar order. In Table III, the bold lines refer to the versions, which, as we could see in the graphs in Figure 10, fared better in terms of runtime, i.e. `basic-row`, `fission-row-ij`, and `fission-col-ji`. We can see for them that running them on 80 threads causes an increase in the runtime (values below 1), contrary to expectations. However, we can also notice that some algorithms speed up when they are run on 80 threads (values above 1) but this applies to versions which, as we could see from the graphs in Figure 10, had a longer runtime. For those versions of the algorithm that perform better in terms of runtime, the machine parameters are sufficient, therefore running HT for them does not improve their runtime. On the other hand, for those that perform weaker in terms of runtime, the capabilities of the machine are not sufficient, therefore HT improves the runtime.

Although 80 threads give these versions some speedup, they still perform worse in runtime than `basic-row`, `fission-row-ij` or `fission-col-ji`. We can see that

the versions with the shortest runtimes, i.e. `basic-row`, `fission-row-ij`, and `fission-col-ji`, perform better in terms of time on 40 threads. Therefore, we will conduct further tests only for the algorithm operating on 40 threads. We will carry out further considerations by selecting one version of the basic algorithm and one version of the fission algorithm. In the case of the basic version of the algorithm, we will choose the one for which we had a better runtime, i.e. `basic-row`.

However, in the case of fission versions, we have two versions, the runtime of which is better than the others and comparable to `fission-row-ij` and `fission-col-ji`, that is, those where the loop order was consistent with the matrix layout, therefore it is not surprising that they perform similarly. For further considerations, we will decide on any one of them, e.g. `fission-row-ij`, with the same matrix representation — row-wise — as for the chosen basic version.

Of the two algorithm versions selected for further tests, `basic-row` tends to perform slightly better than `fission-row-ij` in terms of runtime for different data sizes (at up to 12%).

C. Energy Consumption. Matrix layout and loop interchange

The graph from Figure 11 shows the energy consumption in joules for the `basic-row` (red) and `fission-row-ij` (blue) versions for all four data sizes.

We can see that less energy is consumed by the `basic-row` algorithm, we see this for each data size but it is a small amount between 1% and 11% percent. (respectively for data sizes: 10.33%, 1.33%, 10.98%, 2.63%). We can also observe that as the data size increases, the energy consumption also varies proportionally, and it is the following increase:

$$c_e(k) = k^{\frac{3}{2}}$$

where k is the data size growth factor in our case equal respectively: 1, 2.25, 4 and 16.

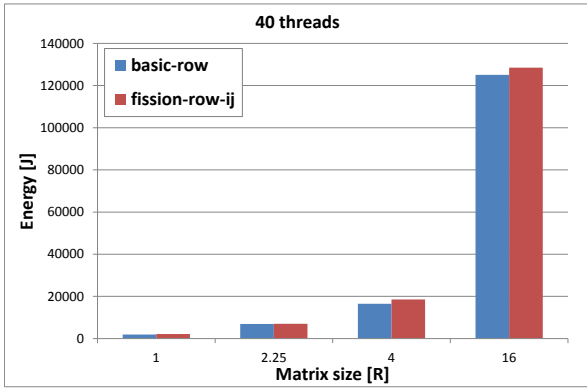


Fig. 11: Energy consumption for `basic-row` and `fission-row-ij` algorithms for different data sizes.

In general, energy consumption is to be expected proportional to the number of operations performed, and this is proportional to $k^{\frac{3}{2}}$.

TABLE IV: Energy consumption ratios as the data size increases.

	R	2.25R	4R	16R
<code>basic-row</code>	1	3.59	8.54	64.72
<code>fission-row-ij</code>	1	3.26	8.6	59.6
$c_e(k)$	1	3.37	8	64

Table IV shows how many times the energy consumption of individual versions has increased for each data size in relation to the energy consumption for the data size R.

D. Execution Time. Strip-mining and loop tiling

Now we will consider the block version of WZ factorization with strip-mining and loop tiling. We will only consider the algorithm implementations that gave the best results in the tests from the previous sections, limiting ourselves to two of them: `basic-row` and `fission-row-ij`. The effect of different `BLOCK_SIZE` on the efficiency will be tested experimentally. We will receive the following versions of the algorithm for further testing:

- `basic-row-sm-b`
- `fission-row-ij-lt-b`

The following block sizes (b) will be tested: 8, 16, 32, 64, 128, 256, 512. In total, 14 different versions of the algorithm will be tested. Our goal is to experimentally investigate which block size will work best in terms of the algorithm's runtime.

In Figure 12 we have graphs showing the runtime of the `basic-row-sm-b` versions for different block sizes b . Each graph deals with the operation of the algorithm on a different size of data. We can see that for individual data sizes (that is: R, 2.25R, 4R, 16R), the block algorithm is the fastest for the block sizes: 64, 64, 256 and 256, respectively, and the slowest

for the block sizes: 8, 16, 8 and 8, respectively. The summary of these observations is presented in Table V.

In Table V the columns present information about the algorithm operating on the specified data size, that is, R, 2.25R, 4R, and 16R, respectively. The last line shows the percentage profit between the slowest and the fastest version of the algorithm, i.e. the profit resulting from the selection of the best-performing block size for the `basic-row-sm-b` versions of the algorithm and the specified data size.

Summarizing the data collected in Table V, we can see that we cannot clearly indicate one block size that would give equally good results for all data sizes. One can notice that for smaller data sizes: (R, 2.25R), the smaller block works better, while for larger data sizes (4R, 16R), the larger block works better. However, we can clearly see which block size perform the worst, these are smaller block size. We can then conclude that small blocks work poorly.

TABLE V: The best and the worst block size due to the `basic-row-sm-b` versions runtime for different data sizes.

	R	2.25R	4R	16R
min. time [s]	8.67	28.90	68.54	556.17
The best block size	64	64	256	256
max. time [s]	11.26	32.53	85.32	601.59
The worst block size	8	16	8	8
max-mix [s]	2.58	3.63	16.78	45.42
%	23%	11%	20%	8%

Let's see what the situation looks like for the `fission-row-ij-lt-b` versions of the algorithm. In Figure 13 we have graphs showing the runtime of the `fission-row-ij-lt-b` versions for different block sizes. Here each graph deals with the operation of the algorithm on a different size of data too. We can see that for individual data sizes the algorithm is the fastest for the block sizes: 8, 8, 64, and 64, respectively, and the slowest for the block sizes: 512, 512, 8, and 8, respectively. The summary of these observations is presented in Table VI.

TABLE VI: The best and the worst block size due to the `fission-row-ij-lt-b` versions runtime for different data sizes.

	R	2.25R	4R	16R
min. time [s]	11.00	34.03	89.69	681.83
The best block size	8	8	64	64
max. time [s]	18.77	48.78	198.51	1904.31
The worst block size	512	512	8	8
max-mix [s]	7.77	14.75	108.82	1222.48
%	41%	30%	55%	64%

Looking at Figures 12, 13 and Tables V, VI, we can see that the `basic-row-sm-b` versions of the algorithm is better in terms of runtime than the `fission-row-ij-lt-b` versions, regardless of the selection of the block size. In the case of the `fission-row-ij-lt-b` versions of the algorithm, we can no longer conclude that, in general, regardless of the size of the data, small blocks perform worse. However, we

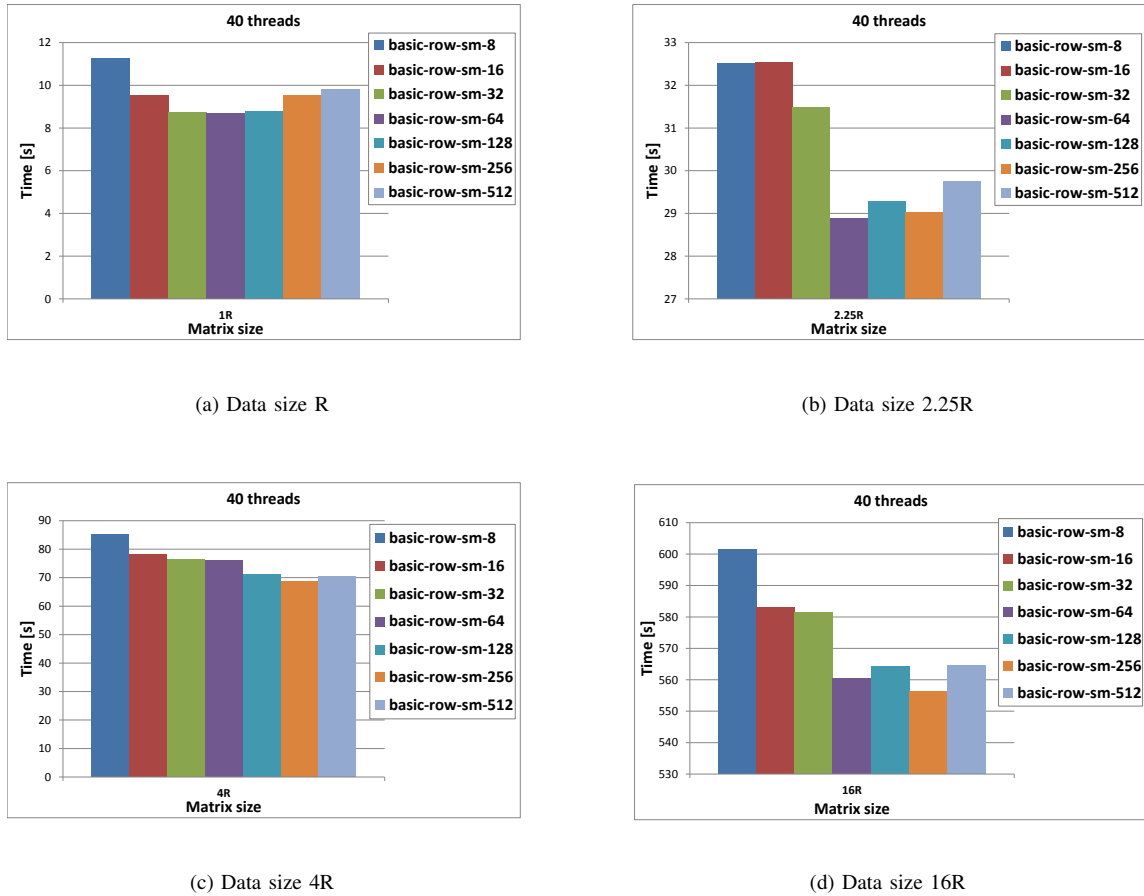


Fig. 12: Execution time of `basic-row-sm-b` versions for different data sizes and different block sizes.

can still observe that as the data size increases, the block size that works best also grows.

From the tables above, we can see that choosing the correct block size can save up to 23% of the time of the `basic-row-sm-b` versions of the algorithm (Table V), and even 64% in the case of the `fission-row-ij-lt-b` versions of the algorithm (Table VI).

E. Energy Consumption. Strip-mining and loop tiling.

Tables VII and VIII provide a summary of the energy consumption during the operation of the various versions of the tested algorithms. The last line shows the percentage energetic profit between the most and the least energy-consuming version of the algorithm, i.e. the profit resulting from the selection of the best-performing block size for the algorithm and the specified data size. Accordingly, Table VII presents data for the `basic-row-sm-b` versions of the algorithm and Table VIII for the `fission-row-ij-lt-b` versions.

We can see that choosing the right block is very important because it can save us 22% of energy consumption in the case of the `basic-row-sm-b` version of the algorithms (Table VII) and up to 61% of energy consumption in the case of the `fission-row-ij-lt-b` versions of the algorithm (Table VIII).

TABLE VII: The best and the worst block size due to energy consumption of the `basic-row-sm-b` algorithm for different data sizes.

	R	2.25R	4R	16R
min [J]	1687.53	6152.07	14925.4	118442.00
The best block size	64	64	256	256
max [J]	2159.86	6827.51	18542.5	129889.00
The worst block size	8	16	8	8
max-mix [J]	472.32	675.44	3617.03	11447.00
%	22%	10%	20%	9%

TABLE VIII: The best and the worst block size due to energy consumption of the `fission-row-ij-lt-b` algorithm for different data sizes.

	R	2.25R	4R	16R
min [J]	2189.55	7180.41	18640.1	143575.00
The best block size	32	8	64	64
max [J]	3863.04	10154.80	39273.60	370432.00
The worst block size	512	512	8	8
max-mix [J]	1673.49	2974.36	20633.60	226858.00
%	43%	29%	53%	61%

F. Time execution-energy trade-off

Although this is usually the case, the best runtime does not always result in the best energy consumption. We can see it

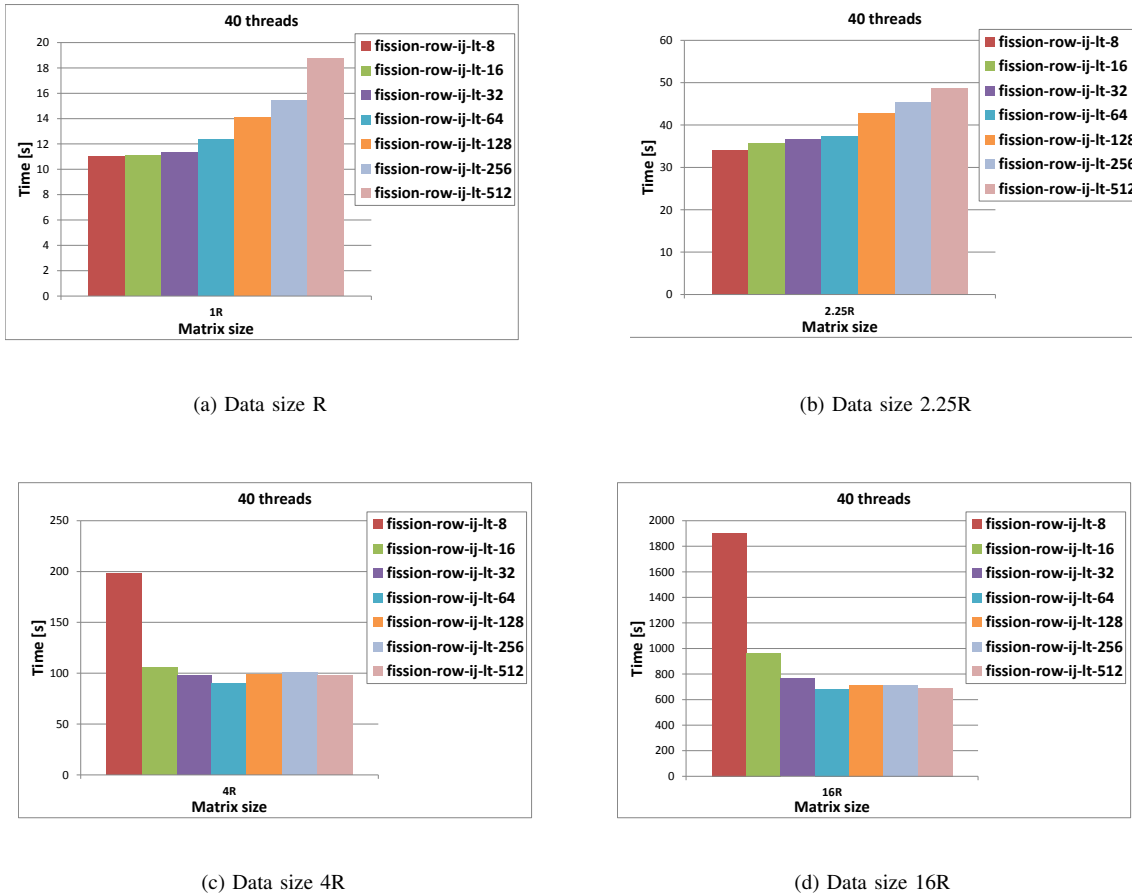


Fig. 13: Execution time of `fission-row-ij-lt-b` versions for different data sizes and different block sizes.

in Tables VI and VIII which show data from experiments on block versions of the fission algorithm, where for the size R , data block size giving the best result in terms of runtime is 8, while the best in terms of consumption energy proved to be 32.

Finally, we present in Table IX a summary of energy consumption, performance, and energy efficiency of the algorithm versions, which are the best during the experiences, for the largest tested data size (16R).

Analyzing the data in Table IX, we can see that `basic-row-sm-256` is the best among the best versions of the algorithm. In terms of energy efficiency, it is 6% better than the second in line, i.e. `basic-row`. However, for smaller data sizes, it turned out to be the best version `basic-row-sm-64` (see Table V and Table VII).

V. CONCLUSION

This article investigates four loop transformation strategies for the WZ factorization, namely: loop fission, loop interchange (permutation), strip-mining and loop tiling. The loop transformation affects both runtime and energy consumption. It can have both a positive effect in reducing runtime and energy consumption and a negative effect in increasing runtime

and energy consumption. Measurements were made on a 2nd Generation Intel Xeon Scalable Processors using the Intel RAPL interface.

Our experiments have shown that the `basic-row` version is definitely better in terms of runtime than the `basic-col`. The advantage of the former is the greater, the larger the size of the data we process and our tests show that it ranges from 19 to even 60 times faster, it is `basic-row`, we can see it in the graph in Figure 10. The first described experiments also showed that HT does not bring benefits in our case.

Our tests have also shown that the loop interchange transformation we propose has a large impact on the reduction of calculation time and energy consumption. The versions for which the loop interchange was compatible with the matrix representation, i.e. `fission-row-ij` and `fission-col-ji`, perform better in terms of operating time. They fell out the better, the larger the size of the data was processed and our experiments showed that it was from 6 to 16 times faster than in the case of a loop interchange inconsistent with the matrix representation. So we should never choose a loop interchange inconsistent with the matrix representation.

However, when comparing the energy consumption for the `basic-row` and `fission` versions with the loop interchange

TABLE IX: Energy efficiency for four best versions of the algorithm (dataset: 16R).

Versions	Time [s]	Total energy [J]	Performance [Gflops/s]	Energy efficiency[Gflops/J]
basic-row	582.63	125098.72	40.26	0.19
fission-row-ij	588.15	128481.50	39.88	0.18
basic-row-sm-256	556.17	118442.28	42.17	0.20
fission-row-ij-lt-64	681.83	143574.55	34.40	0.16

consistent with the matrix representation `fission-row-ij`, we saw that the `basic-row` version was slightly better from 1% to 11% less energy consumption (Figure 11). So the fission transformation won't pay off.

Finally, our experiments have shown that the best version among block versions depends on the data size, and here block size must be selected experimentally. The only thing we can see is that as the data size increases, the block size also increases. It may turn out that if the block size is poorly selected, the energy consumption may be higher by up to 61%. For 16R data size, they are versions `basic-row-sm-256` and `fission-row-ij-lt-64` which works the best (see Table IX). Moreover, we can say that regardless of the size of the data, the application of the strip-mining transformation worked best. For data size 16R the `basic-row-sm-256` version turned out to be the most profitable, as can be seen in Table IX. On the other hand, the transformation of loop tiling does not pay off because it causes a lot of complications in the code and it gives a slight extension of the runtime and slightly higher energy consumption.

Future work includes extending our experimental comparison to a wide range of architectures, including graphics cards. In addition, we will evaluate the performance impact of various runtime systems for OpenMP configurations and loop transformation energy for the WZ and the three decomposition main kernels in dense linear algebra algorithms (Cholesky, LU, and QR).

REFERENCES

- [1] Beata Bylina and Jarosław Bylina. OpenMP Thread Affinity for Matrix Factorization on Multicore Systems. *Proceedings of the Federated Conference on Computer Science and Information Systems*, 11:489–492, 2017. <https://doi.org/10.15439/2017F231>.
- [2] Beata Bylina and Jarosław Bylina. Nested loop transformations on multi- and many-core computers with shared memory. In *Selected Topics in Applied Computer Science*, volume 1, pages 167–186. Maria Curie-Skłodowska University Press, Lublin, 2021. http://stacs.matrix.umcs.pl/v01/stacs_v01.pdf.
- [3] Simplice Donfack, Jack Dongarra, Mathieu Faverge, Mark Gates, Jakub Kurzak, Piotr Luszczek, and Ichitaro Yamazaki. A survey of recent developments in parallel implementations of Gaussian elimination. *Concurrency and Computation: Practice and Experience*, 27(5):1292–1309, 2014. <https://doi.org/10.1002/cpe.3306>.
- [4] D.J. Evans and M. Hatzopoulos. A parallel linear system solver. *International Journal of Computer Mathematics*, 7(3):227–238, 1979. <https://doi.org/10.1080/00207167908803174>.
- [5] Franz Franchetti, Yevgen Voronenko, and Markus Püschel. Formal loop merging for signal transforms. *SIGPLAN Not.*, 40(6):315–326, June 2005. <https://doi.org/10.1145/1064978.1065048>.
- [6] Fred G. Gustavson. *New Generalized Matrix Data Structures Lead to a Variety of High-Performance Algorithms*, pages 211–234. Springer US, Boston, MA, 2001. https://doi.org/10.1007/978-0-387-35407-1_13.
- [7] D. Hackenberg, R. Schöne, T. Ilsche, D. Molka, J. Schuchart, and R. Geyer. An energy efficiency feature survey of the Intel Haswell processor. In *2015 IEEE International Parallel and Distributed Processing Symposium Workshop*, pages 896–904, 2015. <https://doi.org/10.1109/IPDPSW.2015.70>.
- [8] Vasilios Kelefouras and Karim Djemame. A methodology for efficient code optimizations and memory management. In *Proceedings of the 15th ACM International Conference on Computing Frontiers*, CF '18, page 105–112, New York, NY, USA, 2018. Association for Computing Machinery. <https://doi.org/10.1145/3203217.3203274>.
- [9] K. Khan, M. Hirki, T. Niemi, J. Nurminen, and Z. Ou. RAPL in action: Experiences in using RAPL for power measurements. *ACM Transactions on Modeling and Performance Evaluation of Computing Systems (TOMPECS)*, 3, 01 2018. <https://doi.org/10.1145/3177754>.
- [10] Martin Kong and Louis-Noël Pouchet. Model-driven transformations for multi- and many-core CPUs. In *Proceedings of the 40th ACM SIGPLAN Conference on Programming Language Design and Implementation*, PLDI 2019, page 469–484, New York, NY, USA, 2019. Association for Computing Machinery. <https://doi.org/10.1145/3314221.3314653>.
- [11] João V.F. Lima, Issam Raïs, Laurent Lefevre, and Thierry Gautier. Performance and energy analysis of openmp runtime systems with dense linear algebra algorithms. In *2017 International Symposium on Computer Architecture and High Performance Computing Workshops (SBAC-PADW)*, pages 7–12, 2017. <https://doi.org/10.1109/SBAC-PADW.2017.10>.
- [12] João Vicente Ferreira Lima, Issam Raïs, Laurent Lefevre, and Thierry Gautier. Performance and energy analysis of OpenMP runtime systems with dense linear algebra algorithms. *The International Journal of High Performance Computing Applications*, 33(3):431–443, 2019. <https://doi.org/10.1177/1094342018792079>.
- [13] Maxime Mirka, Guillaume Devic, Florent Bruguier, Gilles Sassatelli, and Abdoulaye Gamatié. Automatic energy-efficiency monitoring of openmp workloads. In *2019 14th International Symposium on Reconfigurable Communication-centric Systems-on-Chip (ReCoSoC)*, pages 43–50, 2019. <https://doi.org/10.1109/ReCoSoC48741.2019.9034988>.
- [14] Louis-Noël Pouchet, Uday Bondhugula, Cédric Bastoul, Albert Cohen, J. Ramanujam, P. Sadayappan, and Nicolas Vasilache. Loop transformations: Convexity, pruning and optimization. *SIGPLAN Not.*, 46(1):549–562, January 2011. <https://doi.org/10.1145/1925844.1926449>.
- [15] Yukinori Sato, Tomoya Yuki, and Toshio Endo. An autotuning framework for scalable execution of tiled code via iterative polyhedral compilation. *ACM Trans. Archit. Code Optim.*, 15(4), January 2019. <https://doi.org/10.1145/3293449>.
- [16] Md Abdullah Shahneous Bari, Abid M. Malik, Ahmad Qawasmeh, and Barbara Chapman. Performance and energy impact of openmp runtime configurations on power constrained systems. *Sustainable Computing: Informatics and Systems*, 23:1–12, 2019. <https://doi.org/10.1016/j.suscom.2019.04.002>.
- [17] Przemysław Stpiczyski. Vectorized algorithm for multidimensional Monte Carlo integration on modern GPU, CPU and MIC architectures. *J. Supercomput.*, 74(2):936–952, February 2018. <https://doi.org/10.1007/s11227-017-2172-x>.
- [18] Aleksandar Vitorović, Milo V. Tomašević, and Veljko M. Milutinović. Chapter five - manual parallelization versus state-of-the-art parallelization techniques: The spec cpu2006 as a case study. In Ali Hurson, editor, *Advances in Computers*, volume 92 of *Advances in Computers*, pages 203 – 251. Elsevier, 2014. <https://doi.org/10.1016/B978-0-12-420232-0.00005-2>.
- [19] P. Yalamov and D.J. Evans. The WZ matrix factorisation method. *Parallel Computing*, 21(7):1111–1120, 1995. [https://doi.org/10.1016/0167-8191\(94\)00088-R](https://doi.org/10.1016/0167-8191(94)00088-R).

A short note on post-hoc testing using random forests algorithm: Principles, asymptotic time complexity analysis, and beyond

Lubomír Štěpánek

Department of Statistics and Probability
 Faculty of Informatics and Statistics
 University of Economics

nám. W. Churchilla 4, 130 67 Prague, Czech Republic
 lubomir.stepanek@vse.cz

&

Institute of Biophysics and Informatics
 First Faculty of Medicine
 Charles University

Salmovská 1, Prague, Czech Republic
 lubomir.stepanek@lf1.cuni.cz

Filip Habarta, Ivana Malá, Luboš Marek

Department of Statistics and Probability
 Faculty of Informatics and Statistics
 University of Economics

nám. W. Churchilla 4, 130 67 Prague
 Czech Republic

{filip.habarta, malai, marek}@vse.cz

Abstract—When testing whether a continuous variable differs between categories of a factor variable or their combinations, taking into account other continuous covariates, one may use an analysis of covariance. Several post-hoc methods, such as Tukey’s honestly significant difference test, Scheffé’s, Dunn’s, or Nemenyi’s test are well-established when the analysis of covariance rejects the hypothesis there is no difference between any categories. However, these methods are statistically rigid and usually require meeting statistical assumptions. In this work, we address the issue using a random forest-based algorithm, practically assumption-free, classifying individual observations into the factor’s categories using the dependent continuous variable and covariates on input. The higher the proportion of trees classifying the observations into two different categories is, the more likely a statistical difference between the categories is. To adjust the method’s first-type error rate, we change random forest trees’ complexity by pruning to modify the proportions of highly complex trees. Besides simulations that demonstrate a relationship between the tree pruning level, tree complexity, and first-type error rate, we analyze the asymptotic time complexity of the proposed random forest-based method compared to established techniques.

I. INTRODUCTION

COMPARING a continuous variable’s means of two or more categories (or their combinations) of one or more factor variables and detecting significant mutual differences, if any, is very common in applied statistics. Particularly when the dependent variable needs to be adjusted by other continuous covariates, an analysis of covariance (ANCOVA) is a tool of choice [1].

Since the analysis of covariance tests whether there is, in general, a difference between at least two categories of a given factor, the big question is to determine where exactly the statistical difference is, i. e., which two (or more) exact

categories of the factor are those the significant difference arises from.

For this reason, post-hoc tests are usually applied to identify the significantly different categories of their combinations. Some of them are quite established, for instance, Tukey honestly significant difference (HSD) test [2], Scheffé’s test [3], Dunn’s test [4], or, if needed, Nemenyi’s test with a reduced amount of assumptions required to be met [5].

However, the covariance analysis and the post-hoc tests are limited by relatively tough statistical assumptions, usually in terms of normality of independence of observation subsamples. Furthermore, empirically, when there are multiple methods for one task, that usually implies each method is limited somehow and, consequently, there is no "apriori first choice" method routinely working in all situations.

This work introduces a new post-hoc method based on a random forest algorithm to overcome the mentioned. Each classification tree the random forest model consists of has got its complexity, i. e. number of leaf nodes, by which it can classify into only one or more categories of observations than only one. The continuous dependent variable and continuous covariates are the variables by which an entry sample of observations is split into subsamples, using logical formulas with the variables and searched cut-offs. Considering the categories given by the factor variable (or more factor variables) entering the analysis of covariance, these may be refined as an output for the random forest algorithm, not only as an input for the analysis of covariance. If the number of trees in the random forest model with sufficient complexity, i. e. classifying into two or more factor categories or their combinations, is high enough, then the hypothesis that there is no statistical difference between the two categories is hardly

likely. As a tuning parameter, the pruning level may affect how complex the trees in a random forest are.

After the well-established methods revisiting, we describe principles behind the random forest-based algorithm for post-hoc testing, derive the asymptotic time complexity of the proposed method, and estimate a feasible number of trees in the random forest model regarding the number of other factors and continuous covariates. Eventually, we do simulations to compare the new method to others, i. e. established ones, and, particularly, describe a relationship between the random forest tree pruning level and tree complexity and the model's first-type error rate.

II. PRINCIPLES AND ASSUMPTIONS OF ANALYSIS OF COVARIANCE AND POST-HOC TESTS REVISITED

In this section, we recapitulate basic principles of analysis of covariance and commonly used post-hoc tests to refresh their logic and mention their assumptions and limitations.

A. Analysis of covariance (ANCOVA) – principles, assumptions, and limitations

Principles of ANCOVA. Analysis of covariance is a linear model standing in between analysis of variance (ANOVA) [6] and linear regression [1], [7]. While the analysis of variance assumes there is a continuous dependent variable and independent categorical factors, linear regression allows for independent covariates as the analysis of covariance. However, compared to the linear regression, it estimates effect sizes as excesses above or below covariate variable average and enables to elegantly estimate an explained variability proportion of the continuous dependent variable by covariates. Furthermore, analysis of variance is performed particularly when continuous covariates are not of much interest compared to the factors. That being said, inference tests for coefficients of the covariates are usually skipped.

A model of the analysis of covariance, including $k \in \mathbb{N}$ categorical factor variables and $m \in \mathbb{N}$ continuous covariates, is for i -th observation from $n \in \mathbb{N}$ observations in total, as follows,

$$y_i = \mu + \sum_{j=1}^k \delta_j + \sum_{l=1}^m \beta_l x_{i,l} + \varepsilon_i, \quad (1)$$

where y_i is a value of the dependent continuous variable for i -th observation, μ is a grand total mean of the dependent variable, δ_j is an effect of j -th factor on i -th observation, with $\forall j \in \{1, 2, \dots, k\}$, β_l is a coefficient (slope) of l -th covariate, with $\forall l \in \{1, 2, \dots, m\}$, $x_{i,l}$ is a value of l -th covariate for i -th observation, and ε_i is a residual term of i -th observation, respectively.

Firstly, coefficients β_l for $\forall l \in \{1, 2, \dots, m\}$, listed in a vector $\boldsymbol{\beta} = (\beta_1, \beta_2, \dots, \beta_m)^T$, are estimated as those

minimizing the sum of residuals [8] from formula (1), ignoring (not yet estimated) effects δ_j of the factor variables, thus,

$$\begin{aligned} \boldsymbol{\beta} &= \arg \min_{\boldsymbol{\beta}_l \in \mathbb{R}^m} \left\{ \sum_{i=1}^n \varepsilon_i^2 \right\} \Big|_{\forall j \in \{1, 2, \dots, k\}: \delta_j = 0} = \\ &= \arg \min_{\boldsymbol{\beta}_l \in \mathbb{R}^m} \left\{ \sum_{i=1}^n \left(y_i - \left(\mu + \sum_{l=1}^m \beta_l x_{i,l} \right) \right)^2 \right\} = \\ &= \arg \min_{\boldsymbol{\beta}_l \in \mathbb{R}^m} \left\{ \sum_{i=1}^n \left(y_i - \mu - \sum_{l=1}^m \beta_l x_{i,l} \right)^2 \right\}. \quad (2) \end{aligned}$$

So far, considering formula (2) the analysis of covariance is similar to the linear regression. Secondly, once the vector $\boldsymbol{\beta} = (\beta_1, \beta_2, \dots, \beta_m)^T$ of linear coefficients is estimated, a part close to multifactorial analysis of variance follows. A total sum of squares, $SS_{\text{tot}} = \sum_{i=1}^n (y_i - \bar{y})^2$, describing total variability of the dependent variable [9], is corrected (reduced) by variability explained by the continuous covariates, SS_{β_l} , getting SS_{tot}^* , so

$$\begin{aligned} SS_{\text{tot}}^* &= SS_{\text{tot}} - SS_{\beta_l} = \sum_{i=1}^n (y_i - \bar{y})^2 - \sum_{l=1}^m \frac{\text{cov}(\mathbf{y}, \mathbf{x}_l)^2}{\text{var}(\mathbf{x}_l)} = \\ &= \sum_{i=1}^n (y_i - \bar{y}^*)^2, \quad (3) \end{aligned}$$

considering a vector of the dependent variable $\mathbf{y} = (y_1, y_2, \dots, y_n)^T$, a vector of l -th covariate $\mathbf{x}_l = (x_{1,l}, x_{2,l}, \dots, x_{n,l})^T$, and grand mean \bar{y}^* adjusted by the correction.

Finally, k -way analysis of variance for the coefficients δ_j estimation is applied. The null hypothesis claiming that j -th factor does not affect the dependent continuous variable, i. e. $H_0 : \delta_j = 0$, is formulated k -times and tests using the adjusted sum of squares, SS_{tot}^* , decomposed into component for the factors and for the residuals. Using formula (3) and assuming j -th factor has got n_j categories and c -th category has got $n_{j,c}$ observations, the decomposition is as follows [9],

$$\begin{aligned} SS_{\text{tot}}^* &= \sum_{i=1}^n (y_i - \bar{y}^*)^2 = \sum_{j=1}^k \sum_{c=1}^{n_j} \sum_{i=1}^{n_{j,c}} (y_i - \bar{y}^*)^2 = \\ &= \sum_{j=1}^k \sum_{c=1}^{n_j} \sum_{i=1}^{n_{j,c}} (y_i - \bar{y}_{j,c} + \bar{y}_{j,c} - \bar{y}^*)^2 = \\ &= \sum_{j=1}^k \sum_{c=1}^{n_j} \sum_{i=1}^{n_{j,c}} (y_i - \bar{y}_{j,c})^2 + \sum_{j=1}^k \sum_{c=1}^{n_j} (\bar{y}_{j,c} - \bar{y}^*)^2 = \\ &= \sum_{j=1}^k SS_{j\text{-th factor}} + SS_{\varepsilon} \quad (4) \end{aligned}$$

where $SS_{j\text{-th factor}}$ is sum of squares for j -th factor, SS_{ε} is sum of squares for residuals, and $\bar{y}_{j,c}$ is an average of all values that belong to c -th category of j -th factor. Consequently, using

formula (4), the null hypothesis $H_0 : \delta_j = 0$ for j -th factor is rejected on confidence level $1 - \alpha$ if and only if

$$F = \frac{SS_{j\text{-th factor}}/(n_j - 1)}{SS_{\varepsilon}/(n + k - 1 - \sum_{j=1}^k n_j)} \geq F_{1-\alpha} \left(n_j - 1, n + k - 1 - \sum_{j=1}^k n_j \right), \quad (5)$$

where $F_{1-\alpha}(df_1, df_2)$ is $(1-\alpha)$ -th quantile of Fisher-Snedecor distribution with df_1 and df_2 degrees of freedom, respectively. Rejecting the null hypothesis for j -th factor does not determine which categories of the factor mutually differ significantly, though.

Assumptions and limitations of ANCOVA. Analysis of covariance assumes that residuals are independent, i. e. for each $r, s \in \{1, 2, \dots, n\}$ so that $r \neq s$ is $\text{cov}(\varepsilon_r, \varepsilon_s) = 0$, and of the same variance, i. e. for each $r \in \{1, 2, \dots, n\}$ is $\varepsilon_r = \sigma^2 < 0$. Moreover, the residuals should be normally distributed, i. e. for each $r \in \{1, 2, \dots, n\}$ is $\varepsilon_r \sim \mathcal{N}(0, \sigma^2)$ [1].

B. Post-hoc tests – principles, assumptions, and limitations

Assuming the null hypothesis has been rejected for j -th factor, one would like to determine which exact two or more categories of the factor significantly differ. Let us mark average values of observations that belong to categories c_r and c_s of j -th factor, with $r, s \in \{1, 2, \dots, n_j\}$, as μ_r and μ_s . Usually, the categories c_r and c_s of j -th factor significantly differ when some inequality using data parameters or estimates holds, as showed below applying the mathematical notation from formulas (4) and (5). The decision process may be repeated for each pair of categories r, s of j -th factor to research all possible differences.

1) *Tukey honestly significant differences (HSD) test:* Based on Tukey, averages of the categories c_r and c_s significantly differ if

$$\frac{|\mu_r - \mu_s|}{\hat{\sigma} \sqrt{2/n}} \geq q(\alpha, k, n - k), \quad (6)$$

where $q(\alpha, k, n - k)$ is studentized critical value for confidence level α and σ^2 is residuals' variance, n is sample size and k is the number of factors.

Tukey HSD method assumes that subsamples for compared categories are independent, of the same variability (*homoskedasticity*), and follow normal distribution [2].

2) *Scheffé's test:* Following Scheffé's (unweighted) test, averages of the categories c_r and c_s significantly differ if

$$|\mu_r - \mu_s| \geq \sqrt{(k - 1) \cdot F_{1-\alpha}(df_1, df_2) \cdot SS_{j\text{-th factor}}}, \quad (7)$$

where $df_1 = n_j - 1$ and $df_2 = n + k - 1 - \sum_{j=1}^k n_j$.

Scheffé's test is less limited than Tukey's HSD test since there is no explicit assumption of any normal distribution of observations; however, it has lower statistical power, though [3].

3) *Dunn's test:* Transforming values of dependent variable y_i that belong to the categories c_r and c_s from initial continuous ones to their ranks, we get their averages \bar{w}_r and \bar{w}_s . Dunn's test recommends considering the categories c_r and c_s as different when

$$|\bar{w}_r - \bar{w}_s| \geq z_{1-\alpha/2} \cdot \sqrt{\frac{\frac{n(n+1)}{12} + \sum_{t \in |\bar{w}_r - \bar{w}_s|} \left(n_t^3 - \frac{n_t}{12(n-1)} \right)}{\left(\frac{1}{n_{c_r}} + \frac{1}{n_{c_s}} \right)}}, \quad (8)$$

where t is a possible tied value of the ranks w_r and w_s , n_t is a count of tied ranks at value t , and n_{c_r} and n_{c_s} are numbers of observations in categories c_r and c_s , respectively.

While assumption-free, Dunn's test may fail to identify significant differences between categories due to its low statistical power [4].

4) *Nemenyi's test:* Similarly to Dunn's test, assuming average ranks \bar{w}_r and \bar{w}_s of the categories c_r and c_s , these significantly differ if

$$|\bar{w}_r - \bar{w}_s| \geq q(\alpha, k, n - k) \cdot \sqrt{\frac{n(n+1)}{24} \cdot \left(\frac{1}{n_{c_r}} + \frac{1}{n_{c_s}} \right)}, \quad (9)$$

where $q(\alpha, k, n - k)$ is studentized critical value for confidence level α , n is sample size, k is the number of factors, and n_{c_r} and n_{c_s} are numbers of observations in categories c_r and c_s , respectively.

Nemenyi's test is nonparametric and robust enough, but may suffer from low statistical power, though [5].

III. PRINCIPLES AND ASSUMPTIONS OF THE RANDOM FORESTS

In advance of the proposed method introduction we shortly point out important pieces of knowledge about classification trees and random forests.

A. Classification trees – principles and assumptions

Classification trees from the CART family of trees (classification and regression trees) split a hyperspace of $k \in \mathbb{N}$ explanatory variables (continuous or categorical) into disjunctive hyper-rectangles, fitting simple (constant) models there by minimizing a given criterion [10].

An observation given by a vector of values $\mathbf{x}_i = (x_{i,1}, x_{i,2}, \dots, x_{i,k})^T$ is classified into one of m classes of a target variable by a set of rules that comes from node formulas, created throughout the tree is growing, as described in Fig. 1. Initially, the root covers all observations till a node rule, i. e. a found explanatory variable and a cut-off value minimizing the given criterion partitions the dataset into two parts. Each part is then again split by a new rule set for a child node. The process is recursively repeated by growing the tree, by which a set of node rules successively splits the input dataset into more parts that are mutually more and more different. The process of the tree growing, called a top-down induction of a decision tree (TDIDT), is stopped by a stopping criterion, e. g. maximum of leaf (ending) nodes, maximum tree deepness level, etc.

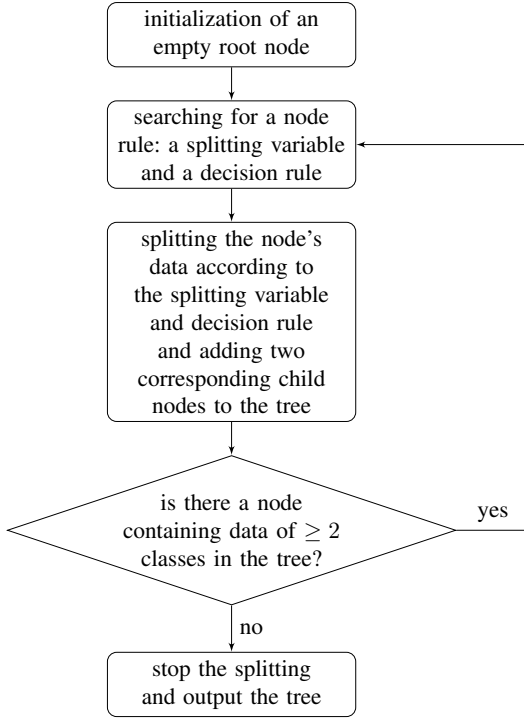


Fig. 1. A top-down induction of a decision tree (TDIDT).

Let $\sigma(\bullet)_j$ be a proportion of all observations that belong – by rules of all nodes from root to leaf one – to a target class j . A leaf node n_t classifies into the class c_f^* if $c_f^* = \operatorname{argmax}_{f \in \{1, 2, \dots, n_j\}} \{\sigma(\bullet)_f\}$. Since each node is through the tree growing a leaf one (for a limited time), the criterion has to be minimized in searching for node n_t rule. There are several commonly used criteria, also called *impurity measure*, $Q_{n_t}(T)$, such as misclassification error (10), Gini index (11), or deviance (cross-entropy) (12),

$$Q_{n_t}(T) = 1 - \sigma(\bullet)_f, \quad (10)$$

$$Q_{n_t}(T) = \sum_{j=1}^m \sigma(\bullet)_j (1 - \sigma(\bullet)_j), \quad (11)$$

$$Q_{n_t}(T) = - \sum_{j=1}^m \sigma(\bullet)_j \cdot \log \sigma(\bullet)_j. \quad (12)$$

One can easily see that the lower the impurity measure is, the higher $\sigma(\bullet)_f$, i. e. a proportion of a target class f in the node n_t , has to be, as expected.

Classification trees, as depicted in Fig. 1, in order to minimize the leaf nodes impurity, tend to overfit the node rules on a given dataset, which is done by the tree's typical "overgrowing", i. e. high complexity. To avoid this, besides some other naive approaches, *pruning* is commonly applied. Firstly, let us use usually defined *cost-complexity function*,

$$C_\kappa(T) = \sum_{n_t \in \{\mathbf{n}_t\}} |\{\mathbf{x}_{n_t}\}| \cdot Q_{n_t}(T) + \kappa \cdot |\{\mathbf{n}_t\}|, \quad (13)$$

where $\{\mathbf{n}_t\}$ is a set of leaf nodes of the tree and $\{\mathbf{x}_{n_t}\}$ is a set of all observations constrained by rules coming from the root till the node n_t . The idea of the pruning is to find a subtree T_κ so that $T_\kappa \subset T$ for a given κ that minimizes the statistics $C_\kappa(T)$, i. e. $T_\kappa = \operatorname{argmin}_T \left\{ \sum_{n_t \in \{\mathbf{n}_t\}} |\{\mathbf{x}_{n_t}\}| \cdot Q_{n_t}(T) + \kappa \cdot |\{\mathbf{n}_t\}| \right\}$.

The $\kappa \geq 0$ is a tuning parameter that governs the trade-off between a high tree complexity and size (for low values of κ) and tree parsimony and reproducibility to other datasets (for large values of κ).

B. Principles of the random forests

Random forests are finite sets of (distinct) classification trees, described in detail above, each classifying a k -dimensional observation, $\mathbf{x}_i = (x_{i,1}, x_{i,2}, \dots, x_{i,k})^T$, into one of $m \in \mathbb{N}$ target classes [11]. The final class $c_f^* \in \{1, 2, \dots, n_j\}$ of a k -dimensional observation is the one the largest subset of the random forest's trees classifies it into¹, i. e., $c_f^* = \operatorname{argmax}_{f \in \{1, 2, \dots, n_j\}} \{\# \text{ of trees classifying into the class } c_f\}$.

What is worth to be mentioned is that only $k^* < k$ variables are considered as possible partitioning variables in node rules. The subset of k^* variables from the original k explanatory variables is selected randomly using bootstrapping; that ensures the pre-selected k^* variables are mutually independent enough. A flowchart of the random forest model building is in Fig. 2.

Neither classification trees nor random forests have important assumptions or limitations worth speaking off.

IV. THE PROPOSED METHOD FOR POST-HOC TESTING

In this section, we introduce a novel alternative for post-hoc testing based on a random forest algorithm. Considering the ANCOVA notation, categories of a factor that contains statistically different effects on the continuous dependent variable are leaf node classes each tree of a random forest classifies into. The dependent variable and the covariates, and other factor variables, if any, are entry variables that serve for node rules if needed. Each tree of the random forest model can either classify only into one category (as a root node tree) or into two or more categories, based on its complexity (size). For details, see Fig. 3.

The more trees of sufficient complexity can classify into the classes (categories) in the forest, the more likely we can reject the null hypothesis that there is no difference between the effects of the factor's categories on the dependent variable. This is formally done by ANCOVA, too. What is more, if a proportion of trees classifying into two given categories is large enough, considering all trees, the given two categories seem to be of statistically different effect on the dependent variable [12].

A proportion of trees classifying into two or more categories to all trees in the random forest is close to a point estimate

¹In case of a tie, i. e. there are two or more target classes the maximum forest's trees classify the observation into, one of them is picked randomly.

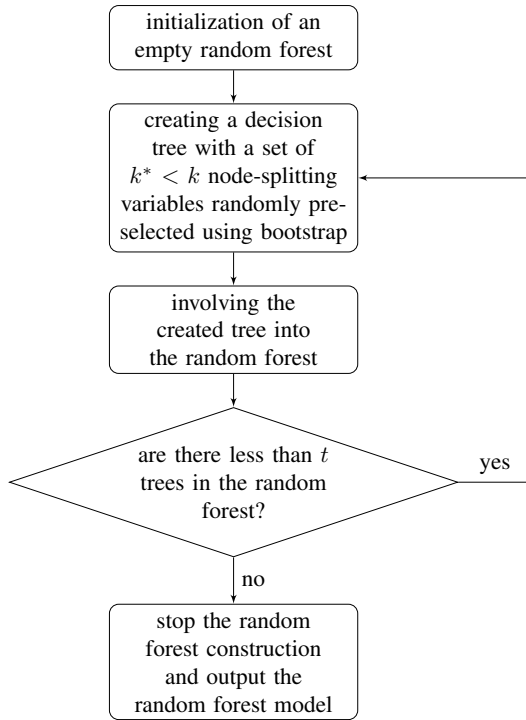


Fig. 2. A construction of the random forest model involving t decision trees.

of the p -value. The p -value is the probability we incorrectly reject the null hypothesis of no different effects of the factor categories on the dependent variable, assuming the null hypothesis is true. Thus, the method also provides statistical inference as a post-hoc test. Since we could modify a random forest's tree complexity (size), i. e. also tendencies to classify either only into one or into two or more classes, by pruning and the tuning parameter κ , we may control the first-type error rate, i. e. the incorrect rejection of the null hypothesis when it is true, of the random forest model as inferential post-hoc test. The proposed method is due to the random forest algorithm behind almost assumption-free.

Besides the derivations of the inferential properties of the method, we also discuss the method's asymptotic time complexity and do a simulation study with varying κ tuning parameters to describe a relationship between the parameter and the method's first-type error rate.

A. Statistical inference behind the proposed method

Using the mathematical notation from previous sections, let us assume the ANCOVA already rejected the null hypothesis that the j -th factor does not affect the dependent variable. Thus, the question is what two (or more) categories of j -th factor are significantly different so that the factor influences

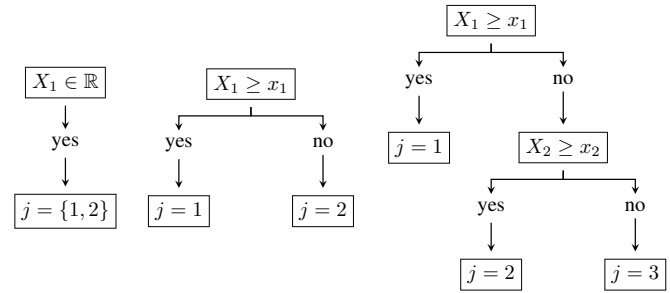


Fig. 3. An example of a root node tree (on the left) not able to classify into any class unambiguously, an example of a tree with sufficient complexity (in the middle) able to classify into two classes ($j = 1$ and $j = 2$), and an example of a tree with sufficient complexity (on the right) able to classify into three classes ($j = 1$, $j = 2$, and $j = 3$).

the dependent variable².

Intuitively, when a large number of the (appropriately pruned) trees of the random forest model can classify into two given classes, i. e. categories, then one can hardly suppose the categories are statistically without a difference.

Similarly to the post-hoc tests, let the null hypothesis H_0 claim that there is no statistical difference between the given two categories c_r and c_s of j -th factor. The alternative hypothesis H_1 claims the contradiction, so

H_0 : No statistical difference between categories c_r and c_s .

H_1 : Statistical difference between categories c_r and c_s .

Whenever a post-hoc test rejects the null hypothesis H_0 in favor of the alternative hypothesis H_1 , the case is equivalent to a situation the test's p -value is lower than or equal to a prior set significance level α , usually equal to 0.05.

By definition, the p -value is a probability of gaining data at least as extreme as the data actually observed, assuming the null hypothesis is true. Let t_c be a number of trees in the random forest model that are in contradiction to the null hypothesis (under the null hypothesis assumption). Then, the value of t_c is equal to the number of all trees classifying, besides other classes, into given two classes (categories) c_r and c_s ; showing that there is a difference between the two classes. Let the $\mathcal{I}_{c_r, c_s}(\tau)$ be an identifier function returning 1 if and only if the tree τ classifies into the classes (categories) c_r and c_s (regardless whether it classifies into other classes, too), thus,

$$\mathcal{I}_{c_r, c_s}(\tau) \begin{cases} 1, & \text{tree } \tau \text{ classifies into categories } c_r \text{ and } c_s, \\ 0, & \text{otherwise.} \end{cases}$$

We can derive

$$t_c = \sum_{\forall \tau \in \text{random forest}} \mathcal{I}_{c_r, c_s}(\tau),$$

²Supposing all categories of j -th factor are similar and mutually without significant differences, then the categorization of j -factor would not result in null hypothesis rejection about no effect of j -th factor on the dependent variable. That is a contradiction, so, there should be two or more different categories of j -th factor.

and assuming the random forest model contains exactly $t \in \mathbb{N}$ trees, and all trees are induced randomly regardless of their complexity³, the p -value is estimated by \hat{p} as

$$\begin{aligned} \hat{p} &= P(\text{getting data at least as extreme as the observed} \mid H_0) = \\ &= P\left(\sum_{\forall \tau \in \text{random forest}} \mathcal{I}_{c_r, c_s}(\tau) \geq t_c \mid H_0\right) = \\ &= P\left(\sum_{\forall \tau \in \text{random forest}} \mathcal{I}_{c_r, c_s}(\tau) \in \{t_c, t_c + 1, \dots, t\}\right) = \\ &= \frac{|\{t_c, t_c + 1, \dots, t\}|}{t} = \\ &= \frac{t - t_c + 1}{t} = \\ &= 1 - \frac{t_c - 1}{t}. \end{aligned} \quad (14)$$

Thus, formula (14) shows that the p -value's estimate is equal to the fraction of $1 - \frac{t_c - 1}{t}$. Intuitively, supposing the initial number t_c of trees in the random forest model that are complex enough to classify, besides others, to categories c_r and c_s is generally low. In that case, such a model is not "much" in contradiction to the null hypothesis about no differences between the two categories. Thus, when the t_c is relatively low, the fraction p -value $= 1 - \frac{t_c - 1}{t}$ is relatively high and close to 1, so, unlikely lower than $\alpha (= 0.05)$. The null hypothesis probably fails to be rejected. However, for high values of t_c , i. e. when there are many trees in the forest with sufficient complexity classifying into the two categories c_r and c_s (thus, in contradiction to the null hypothesis), then – since the high value of t_c – the fraction p -value $= 1 - \frac{t_c - 1}{t}$ is relatively low and perhaps below the α level. Consequently, the null hypothesis is likely rejected.

Indeed, the κ parameter determines how complex the trees in the random forest are or how radical the pruning of the trees is. Investigating formula (13), one can realize that if $\kappa = 0$, then there is no penalization for large tree complexity, so trees in the random forest are generally very complex (i. e., of large size). So, whenever there are at least two observations, one from c_r category and the other from c_s category of j -th factor, all trees in the forest would classify those observations into their categories, i. e. that for each tree τ is $\mathcal{I}_{c_r, c_s}(\tau) = 1$, which results into $t_c = \sum_{\forall \tau \in \text{random forest}} \mathcal{I}_{c_r, c_s}(\tau) = t$, and, thus, p -value estimate is $p\text{-value} = 1 - \frac{t_c - 1}{t} = 1 - \frac{t - 1}{t} = \frac{1}{t} \approx 0$. Finally, if $p\text{-value} \approx 0$, then also $p\text{-value} \approx 0 < \alpha$ which, consequently, results into the null hypothesis rejection. However, when the null hypothesis rejection is often, it is also very likely a *false* rejection, that increases the first-type error rate. High chance of the null hypothesis rejection means also the high statistical power, though, i. e. the case when the incorrect null hypothesis is *correctly* rejected.

For $\kappa > 0$, the penalization for tree complexity (size) is applied, so, the trees' complexity (size) decreases, and, thus, if not all, many of the trees do not classify into both

c_r and c_s categories. This means that there are trees τ in the random forest so that $\mathcal{I}_{c_r, c_s}(\tau) = 0$, and, finally, $t_c = \sum_{\forall \tau \in \text{random forest}} \mathcal{I}_{c_r, c_s}(\tau) < t$. So, p -value estimate is $p\text{-value} = 1 - \frac{t_c - 1}{t} > 0$, and it could be below or above the α level.

B. A feasible low bound of the number of trees t in the random forest

Adopting the ANCOVA mathematical notation, when two categories c_r and c_s of j -th factor are compared using the random forest-based method, other $k - 1$ factors, together with m covariates and the originally dependent continuous variable, play as input variables for node decision rules. Assuming that ℓ -th factor contains $n_\ell \geq 2$ categories and cut-offs for the covariates are usually estimated as midpoints of covariates' ranges, splitting the ranges into a number of categories to be classified into, i. e. n_c , we may estimate a minimum number of mutually different trees. Each mentioned feature could be or could not be included in a tree; that being said, the number of all combinations of $k - 1$ factor node rules is at least $2^{\prod_{\ell \in \{1, 2, \dots, k\} \setminus j} n_\ell} \geq 2^{2^{k-1}}$, and the number of all combinations of m covariates and the dependent variable, split into n_j intervals, is at least n_j^{m+1} . Thus, the minimum number of mutually different trees and, thus, the feasible low bound of the random forest trees' number is

$$t > 2^{\prod_{\ell \in \{1, 2, \dots, k\} \setminus j} n_\ell \cdot n_j^{m+1}} \geq 2^{2^{k-1} \cdot n_j^{m+1}}.$$

Furthermore, since the number of trees t determines decimal precision of p -value estimate based on formula 14, if we ask for decimal precision of $d \in \mathbb{N}$ digits, then the minimum number of trees in the random forest is about

$$t > 10^{d+1},$$

to ensure feasible precision for d -th decimal digit.

C. A brief asymptotic time complexity analysis of the proposed method

The random forest model consists of classification trees as atomic units, constructed following the flowchart 1 and algorithm 1. A decision tree is induced until the moment all its leaf nodes include only observations of one category of j -th factor, i. e. a sequence of node rules coming from root node till the leaf one successively limits the entry dataset to one-class observations [13]. When the categories are well balanced across the dataset, each binary partitioning halves them, and the tree average depth is around $\log_2 n$ levels; thus, the asymptotic time complexity is also $\Theta(\log_2 n)$, assuming one split of a node takes one time atomic unit. However, if the categories are totally unbalanced, each splitting cuts the current dataset of size n^* into 1 and $n^* - 1$ observations, which takes n steps in total. Then, tree depth is n , so the asymptotic time complexity is $\Theta(n)$. Assuming there are $k - 1$ factors, m covariates and the originally dependent variable, i. e. $k - 1 + m + 1 = k + m$ variables, searched through averagely $\frac{n}{2}$ observations within each node splitting, the asymptotic time complexity of one tree induction $\Theta(\dagger)$

³This is ensured by the bootstrapped selection of $k^* < k$ node rule variables.

is therefore between $\Theta(\log_2 n)$ (best-case scenario) and $\Theta(n)$ (worst-case scenario),

$$\begin{aligned} \Theta\left(\frac{(k+m)n}{2} \log_2 n\right) &\leq \Theta(\dagger) \leq \Theta\left(\frac{(k+m)n^2}{2}\right), \\ \Theta\left(\frac{(k+m)n}{2} \log_2 n\right) &\leq \Theta(\dagger) \leq \Theta\left(\frac{(k+m)n^2}{2}\right), \\ \Theta((k+m)n \log_2 n) &\lesssim \Theta(\dagger) \lesssim \Theta((k+m)n^2). \end{aligned} \quad (15)$$

Algorithm 1: The top-down induction of decision trees (TDIDT) following the logic of the flowchart 1

Data: a $n \times (m+k)$ dataset of n observations, with j -th target factor, $k-1$ factors, m covariates, and one dependent continuous variable

Result: a classification tree

```

1  $T = (\{\mathbf{n}\})$  // a tree  $T$  with a set ;
2 // of nodes  $\mathbf{n}$ ;
3  $\{\mathbf{n}\} = \{\text{root}\}$  // initially, the tree  $T$ 
  ;
4 // is a root;
5  $\sigma(\bullet)_j$  // a node criterion;
6 while  $\exists$  a node  $\in \{\mathbf{n}\}$  so that data constrained by all
  node rules coming from root to the node belong to
   $\geq 2$  classes do
7   find for the node a splitting variable and splitting
  point minimizing the  $\sigma(\bullet)_j$ ;
8   add to the node two child nodes  $n_{\text{left}}$  a  $n_{\text{right}}$ ;
9    $\{\mathbf{n}\} := \{\mathbf{n} \cup \{n_{\text{left}}, n_{\text{right}}\}\}$ ;
10   $T := (\{\mathbf{n}\})$  // update the tree using;
11  // the new node set  $\mathbf{n}$ ;
12 end
13 a completely induced tree  $T$ ;
```

Since a random forest contain t trees, each built in $\Theta(\dagger)$ time by (15), the entire random forest asymptotic time complexity construction $\Theta(\ddagger)$ is

$$\begin{aligned} \Theta\left(t \frac{(k+m)n}{2} \log_2 n\right) &\leq \Theta(\ddagger) \leq \Theta\left(t \frac{(k+m)n^2}{2}\right), \\ \Theta(t(k+m)n \log_2 n) &\lesssim \Theta(\ddagger) \lesssim \Theta(t(k+m)n^2). \end{aligned} \quad (16)$$

One model of the random forest provides one (point) estimate of the p -value using the formula (14), enabling to statistically distinguish between two categories. In comparison, the estimation of the decision rules for post-hoc tests using formulas (6), (7), (8), and (9) usually take only several linear steps, assuming the $\text{SS}_{j\text{-th factor}}$ in (7) term is precalculated. Fortunately, the time complexity (16) is still polynomial. Furthermore, since the building of the random forest with the complexity of (16) is based on independent trees induction, it could be parallelized; then, if the random forest building would be parallelized into $\pi \leq t$ independent slave processes each

inducing a bunch of $\frac{t}{\pi} \in \mathbb{N}$ trees, the time complexity (16) would be reduced to

$$\begin{aligned} \Theta\left(t \frac{(k+m)n}{2\pi} \log_2 n\right) &\leq \Theta(\ddagger) \leq \Theta\left(t \frac{(k+m)n^2}{2\pi}\right), \\ \Theta\left(\frac{t}{\pi}(k+m)n \log_2 n\right) &\lesssim \Theta(\ddagger) \lesssim \Theta\left(\frac{t}{\pi}(k+m)n^2\right). \end{aligned}$$

V. SIMULATION STUDY

To compare the established post-hoc tests with the proposed method, particularly its first-type error rate, we run a simulation study generating many $n \times (k+m+1)$ datasets with n observations, k factor variables, m covariates with various relationships between the variables, and, lastly, with the continuous dependent variable. For each post-hoc test, i. e. Tukey HSD test, Scheffé's test, Dunn's test, Nemenyi's test, and random forest-based method, we compare two categories of a selected factor so that the categories have significantly non-different averages within the continuous dependent variable and check how many times the methods claim there is a significant difference. Thus, in theory, we measure the first-type error rate. The simulation was repeated for different κ parameter values to illustrate how the value of κ determines the first-type error rates in the new method, i. e., what are ideal κ values to control the first-type error rate on a feasible level.

The datasets were generated as follows. One of the k factors, let's say the j -th one, contained two categories, c_r and c_s , following a normal distribution with the same average of the continuous dependent variable, i. e. $\mathcal{N}(0, 1^2)$. Other $k-1$ factors always split the dependent continuous variable into 2 to 4 categories with random averages from $\mathcal{N}(\mu, \sigma^2)$, where $\mu \in \langle -1, 1 \rangle$ and $\sigma^2 \in \langle 0, 2 \rangle$. Furthermore, the covariates also followed normal distribution $\mathcal{N}(\mu, \sigma^2)$, where $\mu \in \langle -1, 1 \rangle$, $\sigma^2 \in \langle 0, 2 \rangle$, and correlations \mathbf{r} between the dependent continuous variable and covariates were randomly from $\mathbf{r} \in \langle -0.5, 0.5 \rangle$. The continuous variable was dependent for all post-hoc tests with exception of the proposed method, where j -th factor is as dependent one.

There were $\eta = 1000$ datasets, as depicted above, generated in total, and for each $\kappa \in \{0.1, 0.3, 0.5, 0.7, 0.9\}$. The number of trees in each random forest was always $t = 1000$. Numbers of cases where p -value was lower than or equal to $\alpha = 0.05$ regardless of the post-hoc test were summed up, indicating the point estimates of the first-type error rates, as illustrated in table I. The simulation study was performed using R programming language and environment [14]. There are more numerical applications of R language to various fields in [15]–[23].

While the established post-hoc tests returned point estimates of the first-type error rate about 0.050 (regardless of κ since their formulas (6), (7), (8), and (9) are not functions of the κ), point estimates of the first-type error rates output by the introduced method progressively decreased with increasing value of κ , see table I.

However, since the proposed random forest-based algorithm for categories' averages comparison is data-determined and heuristic, it is nontrivial to suggest specific values of the

TABLE I

POINT ESTIMATES OF THE FIRST-TYPE ERROR RATES FOR POST-HOC TESTS, I. E. TUKEY HSD TEST (T-HSD), SCHEFFÉ'S TEST (SchT), DUNN'S TEST (DT), NEMENYI'S TEST (NT), AND THE PROPOSED METHOD (RF-T) FOR DIFFERENT VALUES OF TUNING PARAMETER κ , BASED ON THE SIMULATION DESCRIBED ABOVE.

	method					κ
	T-HSD	SchT	DT	NT	RF-T	
# of cases in total	1000	1000	1000	1000	1000	0.1
$\#\{p\text{-value} \leq 0.05\}$	56	51	45	42	66	
first-type error rate	0.056	0.051	0.045	0.042	0.066	
# of cases in total	1000	1000	1000	1000	1000	0.3
$\#\{p\text{-value} \leq 0.05\}$	54	51	45	42	58	
first-type error rate	0.056	0.051	0.045	0.042	0.058	
# of cases in total	1000	1000	1000	1000	1000	0.5
$\#\{p\text{-value} \leq 0.05\}$	60	58	45	51	49	
first-type error rate	0.060	0.058	0.045	0.041	0.049	
# of cases in total	1000	1000	1000	1000	1000	0.7
$\#\{p\text{-value} \leq 0.05\}$	50	56	41	50	38	
first-type error rate	0.050	0.056	0.041	0.050	0.038	
# of cases in total	1000	1000	1000	1000	1000	0.9
$\#\{p\text{-value} \leq 0.05\}$	47	53	48	46	29	
first-type error rate	0.047	0.053	0.048	0.046	0.029	

pruning parameter κ to reach a given level of the first-type error rate.

Still, as indicated by the derived theory and simulation study, the higher the pruning parameter κ is, the higher penalization for too complex trees in a random forest is. Thus, the less complex the trees in a random forest are, which results in lower trees' ability to classify into two or more classes of the factor variable and, consequently, the lower first-type error rate of the random forest as an inferential algorithm.

VI. CONCLUSION

When searching for statistical differences between categories' averages of a given factor, once analysis of covariance is performed, post-hoc tests may identify which two or more categories have significantly different impacts on the dependent continuous variable average. However, the post-hoc tests are usually limited by rigid statistical assumptions.

In this work, we introduced a novel method for post-hoc testing based on a random forest algorithm. Rather than a statistical comparison of dependent variable's averages for two factor categories and evaluation of its effect size, the proposed technique refines the logic of testing. The factor with compared categories becomes an output variable, i. e., its categories populate leaf nodes of the model trees, and other factors, initially dependent continuous variable and covariates, if any, serve input variables, i. e., as quantities in node rules. The higher the random forest model trees' complexity, i. e., size is, the more likely the trees classify into (besides others) the two compared categories, and, thus, the null hypothesis claims there is no statistical difference between the compared categories' averages is more likely to be rejected. Furthermore, since trees' pruning level determines the trees in the random forest model complexity, a tuning parameter affecting the significance of the pruning also changes trees' complexity and, consequently, the probability of correctly rejecting the

null hypothesis. Finally, the tree pruning may modify the first-type error rate, too. The asymptotic time complexity of the random forest-based post-hoc method is usually higher than the complexities of the established procedures but is still polynomial and might be parallelized.

Therefore, the introduced random forest-based method seems to be a valid alternative to other, commonly used post-hoc tests.

VII. ACKNOWLEDGEMENT

This paper is supported by the grant OP VVV IGA/A, CZ.02.2.69/0.0/0.0/19_073/0016936 with no. 18/2021, which has been provided by the Internal Grant Agency of the Prague University of Economics and Business.

REFERENCES

- [1] Geoffrey Keppel and Thomas D Wickens. *Design and analysis*. en. 4th ed. Upper Saddle River, NJ: Pearson, Jan. 2004.
- [2] John W. Tukey. "Comparing Individual Means in the Analysis of Variance". In: *Biometrics* 5.2 (June 1949), p. 99. DOI: 10.2307/3001913. URL: <https://doi.org/10.2307/3001913>.
- [3] H Scheffe. *The analysis of variance*. en. Wiley Classics Library. Nashville, TN: John Wiley & Sons, Feb. 1999.
- [4] Olive Jean Dunn. "Multiple Comparisons among Means". In: *Journal of the American Statistical Association* 56.293 (Mar. 1961), pp. 52–64. DOI: 10.1080/01621459.1961.10482090. URL: <https://doi.org/10.1080/01621459.1961.10482090>.
- [5] Myles Hollander and Douglas Alan Wolfe. *Nonparametric Statistical Methods*. en. 2nd ed. Wiley series in probability & statistics: applied section. Nashville, TN: John Wiley & Sons, Feb. 1999.
- [6] Ellen R Girden. *ANOVA: Repeated measures*. 84. Sage, 1992. ISBN: 0803942575.
- [7] Kenneth L. Lange, Roderick J. A. Little, and Jeremy M. G. Taylor. "Robust Statistical Modeling Using the t Distribution". In: *Journal of the American Statistical Association* 84.408 (Dec. 1989), p. 881. DOI: 10.2307/2290063. URL: <https://doi.org/10.2307/2290063>.
- [8] A. Charnes, E. L. Frome, and P. L. Yu. "The Equivalence of Generalized Least Squares and Maximum Likelihood Estimates in the Exponential Family". In: *Journal of the American Statistical Association* 71.353 (Mar. 1976), pp. 169–171. DOI: 10.1080/01621459.1976.10481508. URL: <https://doi.org/10.1080/01621459.1976.10481508>.
- [9] R A Bailey. *Cambridge series in statistical and probabilistic mathematics: Design of comparative experiments series number 25*. Cambridge, England: Cambridge University Press, Apr. 2008.
- [10] Leo Breiman. *Classification and regression trees*. New York: Chapman & Hall, 1993. ISBN: 9780412048418.

- [11] Leo Breiman. “Random Forests”. In: *Machine Learning* 45.1 (2001), pp. 5–32. DOI: 10.1023/a:1010933404324. URL: <https://doi.org/10.1023/a:1010933404324>.
- [12] Lubomír Štěpánek, Filip Habarta, Ivana Malá, and Luboš Marek. “Analysis of asymptotic time complexity of an assumption-free alternative to the log-rank test”. In: *Proceedings of the 2020 Federated Conference on Computer Science and Information Systems*. IEEE, Sept. 2020. DOI: 10.15439/2020f198. URL: <https://doi.org/10.15439/2020f198>.
- [13] Kawther Hassine, Aiman Erbad, and Ridha Hamila. “Important Complexity Reduction of Random Forest in Multi-Classification Problem”. In: *2019 15th International Wireless Communications Mobile Computing Conference (IWCMC)*. 2019, pp. 226–231. DOI: 10.1109/IWCMC.2019.8766544.
- [14] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2017. URL: <https://www.R-project.org/>.
- [15] Lubomír Štěpánek, Pavel Kasal, and Jan Měšťák. “Evaluation of facial attractiveness for purposes of plastic surgery using machine-learning methods and image analysis”. In: *2018 IEEE 20th International Conference on e-Health Networking, Applications and Services (Healthcom)*. IEEE, Sept. 2018. DOI: 10.1109/healthcom.2018.8531195. URL: <https://doi.org/10.1109/healthcom.2018.8531195>.
- [16] Lubomír Štěpánek, Pavel Kasal, and Jan Měšťák. “Machine-learning at the service of plastic surgery: a case study evaluating facial attractiveness and emotions using R language”. In: *Proceedings of the 2019 Federated Conference on Computer Science and Information Systems*. IEEE, Sept. 2019. DOI: 10.15439/2019f264. URL: <https://doi.org/10.15439/2019f264>.
- [17] Lubomír Štěpánek, Pavel Kasal, and Jan Měšťák. “Machine-Learning and R in Plastic Surgery – Evaluation of Facial Attractiveness and Classification of Facial Emotions”. In: *Advances in Intelligent Systems and Computing*. Springer International Publishing, Sept. 2019, pp. 243–252. DOI: 10.1007/978-3-030-30604-5_22. URL: https://doi.org/10.1007/978-3-030-30604-5_22.
- [18] Lubomír Štěpánek, Pavel Kasal, and Jan Měšťák. “Machine-learning at the service of plastic surgery: a case study evaluating facial attractiveness and emotions using R language”. In: *Proceedings of the 2019 Federated Conference on Computer Science and Information Systems*. IEEE, Sept. 2019. DOI: 10.15439/2019f264. URL: <https://doi.org/10.15439/2019f264>.
- [19] Lubomír Štěpánek, Pavel Kasal, and Jan Měšťák. “Evaluation of Facial Attractiveness after Undergoing Rhinoplasty Using Tree-based and Regression Methods”. In: *2019 E-Health and Bioengineering Conference (EHB)*. IEEE, Nov. 2019. DOI: 10.1109/ehb47216.2019.8969932. URL: <https://doi.org/10.1109/ehb47216.2019.8969932>.
- [20] Lubomír Štěpánek, Filip Habarta, Ivana Malá, Luboš Marek, and Filip Pazdírek. “A Machine-learning Approach to Survival Time-event Predicting: Initial Analyses using Stomach Cancer Data”. In: *2020 International Conference on e-Health and Bioengineering (EHB)*. IEEE, Oct. 2020. DOI: 10.1109/ehb50910.2020.9280301. URL: <https://doi.org/10.1109/ehb50910.2020.9280301>.
- [21] Lubomír Štěpánek, Filip Habarta, Ivana Malá, Luboš Marek, and Filip Pazdírek. “A random forest-based approach for survival curves comparing: principles, computational aspects and asymptotic time complexity analysis”. In: *Proceedings of the 16th Conference on Computer Science and Intelligence Systems*. IEEE, Sept. 2021. DOI: 10.15439/2021F89.
- [22] Lubomír Štěpánek, Filip Habarta, Ivana Malá, Luboš Marek, and Filip Pazdírek. “Data envelopment analysis models connected in time series: A case study evaluating COVID-19 pandemic management in some European countries”. In: *2021 International Conference on e-Health and Bioengineering (EHB)*. Iasi, Romania: IEEE, Nov. 2021. DOI: 10.1109/EHB52898.2021.9657597.
- [23] Owen Jones, Robert Maillardet, and Andrew Robinson. *Introduction to Scientific Programming and Simulation Using R*. Chapman and Hall/CRC, Mar. 2009. DOI: 10.1201/9781420068740. URL: <https://doi.org/10.1201/9781420068740>.

30th International Symposium on Concurrency, Specification and Programming

THE symposium on Concurrency, Specification, and Programming is the series of meeting formerly organized every even year by Humboldt University of Berlin and every odd year by Warsaw University. It deals with formal specification of concurrent and parallel systems, mathematical models for describing such systems, and programming and verification concepts for their implementation. The symposium has a tradition dating back to the mid-seventies; since 1993 it was named CS&P. During the past 30 years, CS&P has become an important forum for researchers from European and Asian countries.

TOPICS

The list of topics includes, but is not limited to:

- Mathematical models of concurrency
- Formal specification languages
- Theory of programming
- Model checking and testing
- Multi-agent systems
- Rough sets
- Verification
- Formal aspects of knowledge management
- Knowledge discovery and data mining
- Soft computing
- Applications, e.g. in Robotics

TECHNICAL SESSION CHAIRS

- **Czaja, Ludwik**, Vistula University, Poland
- **Nguyen, Hung Son**, University of Warsaw, Poland
- **Schlingloff, Holger**, Humboldt University, Germany
- **Vogel, Thomas**, Humboldt University, Germany

PROGRAM COMMITTEE

- **Artiemjew, Piotr**, University of Warmia and Mazury, Poland
- **Dutta, Soma**, University of Warmia and Mazury in Olsztyn, Poland
- **Gomolinska, Anna**, University of Bialystok, Institute of Informatics, Poland
- **Redziejowski, Roman**, Poland
- **Senyurek, Edip**, Vistula University, Poland
- **Skowron, Andrzej**, Systems Research Institute, Polish Academy of Sciences and Cardinal Stefan Wyszyński University, Warsaw, Poland
- **Stepaniuk, Jaroslaw**, Bialystok University of Technology, Poland
- **Suraj, Zbigniew**, Institute of Computer Science, College of Natural Sciences, University of Rzeszów, Poland
- **Szczuka, Marcin**, Institute of Informatics, University of Warsaw, Poland
- **Wasilewski, Piotr**, University of Warsaw, Poland
- **Werner, Matthias**, TU Chemnitz, Germany
- **Wolf, Karsten**, Universität Rostock, Germany

An observation on pure strategies in Security Games

Marek Adrian

Department of Applied Computer Science
AGH University of Science and Technology
Al. A. Mickiewicza 30, 30-059 Kraków, Poland
Email: madrian@agh.edu.pl

Gianluigi Greco

DeMaCS Dept., University of Calabria
Via Pietro Bucci, 87036 Arcavacata,
Rende (CS), Italy
Email: ggreco@mat.unical.it

Abstract—Security Games have been used in several different fields to randomise the division of limited resources and thus maximise the possibility of securing a set of targets. For this very practical purpose it is natural to consider primarily mixed strategies, but such focus omits some theoretical properties of the games discussed. In this paper we discuss the existence and properties of pure Nash equilibria in security games. We give an overview of the basic observations that can be made in this setting. We also recognize an interesting problem in a case with multiple players playing a security game asynchronously, propose an algorithm for finding a strategy for any given player in the mentioned case and prove that the strategy profile resulting from the algorithm is in fact a Nash equilibrium and, even stronger, a subgame perfect equilibrium. We think that these findings are a nice supplement of the practical approach to Security Games and allow to form new research questions.

I. INTRODUCTION

Since its conception in the previous century, Game Theory provided a language which has been used to discuss, among other things, how businesses compete on a given market, how to model predator-pray interactions in the animal kingdom and, most famously, how to behave during an interrogation. This should come at no surprise as, at its core, game theory describes conflict between autonomous entities and such beings can be recognized in almost any setting.

One of the most basic concepts that was necessary to define in this field from the very beginning, is how to recognize whether any decision made by an entity is good or not. Typically it is decided by considering the potential outcome, which is provided by knowing the decision process of all of the players, and testing whether it has a desired set of properties.

One of the most known descriptions of a good strategy profile is the Nash equilibrium [1]. While it has its own drawbacks, it provides a reasonable set of assumptions on the behaviour of the players and always exists in finite games with mixed strategies.

A mixed strategy is when the decision process of a player is given by a probabilistic function over the possible actions, and thus is great to describe the uncertainty in decision making. In contrast we have pure strategies, where a player chooses one action. There is no guarantee that there is a Nash equilibrium consisting of only of pure strategies, which makes it a very interesting decision question, that has been studied for several classes of games [2] [3].

An interesting type of game, for which pure strategies were not considered, are the so called security games. Based on an

idea by Stackelberg [4], these games divide the players into two groups, of which one declares their strategies before the other, and tries to find a Nash equilibrium in such setting. This approach has been successfully applied in several systems, like the security checkpoint schedule in LAX airport [5], planning US Air Marshals flight security patterns [6], preventing poaching [7] and other cases. While it should be obvious why limiting yourself to just pure strategies in security games is not the best thing to do from a practical point of view, we found the theoretical properties of this model to be interesting and this paper, which is based on a PhD thesis [8], provides some insights to the approach.

The structure of this paper is the following: In section 2 we provide the basic definitions needed for the discussion and observations that can be made about pure strategies in security games. In next section we discuss deeper a specific case where we have multiple defenders playing asynchronously and recognize which game theory concepts should be used to find the strategies in that setting. Section 4 contains the algorithm for choosing a strategy for each player and the proof that the resulting strategy profile is a subgame perfect equilibrium. In the final section we will briefly summarize our observations and recognize further possible areas of inquiry.

II. BASIC DEFINITIONS AND OBSERVATIONS

As a *game in normal form* we recognise a tuple (N, A, u) where: N is a finite set of players, indexed by i ; $A = A_1 \times \dots \times A_n$ where A_i is a finite set of actions available to player i . Each vector (a_1, \dots, a_n) in A is called an *action profile*; $u = (u_1, \dots, u_n)$, where $u_i : A \rightarrow \mathbb{R}$ is a *valuing function* for player i . In our cases we will assume that the goal of each player is to get the largest possible value from their valuing function. Moreover in the games we will be discussing any action will be corresponding to choosing an object to protect and thus we sometimes use the expressions *committing to an action* and *picking an object* interchangeably, hopefully not causing too much confusion. The assignment of probabilities by a player to his set of available actions is called a *strategy* and the decision on the probabilities is called *committing to a strategy*. If the assignment gives one of the actions probability 1 then it is called a *pure strategy*. Any other strategy is called *mixed*. A *strategy profile* is a vector containing strategies of all players in a given game. To formalize the concepts of good strategies we define a *best response* for the player i to an action

vector a_{-i} (an action vector without the i -th position) is an action $a \in A_i$ such that for all a' in A_i we have $u_i(a_{-i}, a) \geq u_i(a_{-i}, a')$. A strategy $a = (a_1, \dots, a_n)$ is a *Nash equilibrium* iff for all i in $\{1, \dots, n\}$ a_i is a best response to a_{-i} .

Now we can move on to define security games, in which we divide the players in to two groups, one of which declares their strategies before the other. We call the first group the defenders and the second group is called the attackers. In any case an action done by a defender is interpreted as defending a specific object, corridor, monument etc. and an action chosen by the attacker is assaulting it. For it to be a security game, the valuation function has to have an additional property: if the attacker chose to attack an undefended target his valuation function will be higher than if the target was defended. Symmetrically the valuation function for the defender should be worse when an undefended object has been attacked, then if a defended object has been attacked.

While the properties of the valuation function play a crucial role in finding mixed strategies in Security Games, there are not that important when we assume that only pure strategies are available to the players. In the case of pure strategies in security games it is sufficient to just consider the strategies of the defenders. Now let us think what will happen, if all of the defenders have to commit to a strategy at the same time. Obviously, they want their most valued object to be protected and they have no way to coordinate with other players. If for each of them the most valuable object is different, then we will have a Nash equilibrium.

If two players value one object the most we have an interesting situation: on one hand if they are rational they should pick the most valued object, but that will lead to a strategy profile that is not a Nash equilibrium, as if only one of them would switch to his second best object, he would increase his valuation function. On the other there is no rational way for any of those players to make a different decision as they are risking lesser value if both of them choose their second best option. So we would have a situation where there may exist a pure Nash equilibrium, but there is no way for the players to achieve it. This problem could disappear, if the defenders themselves played in a given order, but it may not be the case, and will be the topic of our next inquiry.

III. MULTIPLE DEFENDERS IN AN ASYNCHRONOUS GAME

Let us consider a security game with n defenders and one attacker. Each action of a defender consists of picking an object to defend. The defenders commit to their pure strategies in a given order. The valuation function for each player is given as the sum of all values of the objects picked by all of the players. This model describes a sequential game which can be described as a game in Extended Form. Full formal definitions of an Extended Form game, Nash equilibrium and a sub-game perfect equilibrium in such games can be found in handbooks like [9] or [10]) We will describe the basic intuitions behind those concepts.

We can represent a game in Extended Form as a tree in which: each vertex represents the state of the game at the

moment, the root being the game before any move was made, and each leaf describing each possible outcome of the game, each edge represents an action that the current player can choose and connects the vertex corresponding to the game state before that action to the vertex with the game state after that action. With this representation any sub-tree that starts in a vertex and consists of all the edges and vertices below is also a game and is called a *sub-game*.

Any game in extended form can be translated into normal form and thus we can use the definitions of Nash equilibrium and best response in this context. There is a problem however, as the Nash equilibrium does not have to be optimal on sub-games. A *sub-game perfect equilibrium* is a Nash equilibrium that is also a Nash equilibrium on all of their sub-games.

Now if we have the complete game tree, it is easy to see that we can find a best response for any player simply by backtracking the expected results from the leaves to the current situation. This is unfeasible, as the whole game tree will grow exponentially with respect to the number of players and possible actions. To get rid of this problem, instead of trying to find a whole strategy, we will try to identify a good move and argue that there exists a sub-game perfect equilibrium in which this was the best response.

Consider a game G with the set of actions A and a strategy profile s for G . A sequence $(a_1, \dots, a_n) \in A^n$ is a *result* of strategy s if, and only if starting in the root of the game tree and moving down an edge only if it is the action indicated by the strategy s , the actions the edges traveled through form the sequence (a_1, \dots, a_n) .

We say a sequence of actions (a_1, \dots, a_n) is called *reasonable* if there exists a strategy profile that is a sub-game perfect equilibrium, such that (a_1, \dots, a_n) is a result of s .

Thus we can simplify our problem and instead of finding a strategy profile which should describe actions taken in any possible situation, just find an sequence of actions and argue that they are a result of a good strategy profile.

IV. MAIN RESULT

We will present now the algorithm for finding good move for each player. The algorithm in itself is fairly simple and easily works in polynomial time.

A. Algorithms for decisions

The basic algorithm

Input: A - set of available actions; V - the valuation matrix; i - the index of the player making the decision;

Output: (a_i, \dots, a_n) the predicted choices of actions for players i to n .

- 1) Delete all columns for actions that have already been chosen.
- 2) Define k as the number of rows in the matrix.
- 3) Find in the last row the column in which there is the most valuable object for the k -th player (if more than one pick at random).
- 4) Mark this object as a_k .
- 5) Remove the last row from the matrix.

6) Repeat steps 1-4 until a_i is defined.

The modified algorithm

Input: A - set of available actions;

V - the valuation matrix;

i - the index of the player making the decision;

(a_1, \dots, a_n) - sequence of choices of actions predicted by the original algorithm;

Output: (a'_i, \dots, a'_n) the predicted choices of actions for players i to n .

- 1) Delete all columns for objects that have already been chosen.
- 2) Define k as the number of rows in the matrix.
- 3) Find in the last row the column in which is the most valuable object for the k -th player (if more than one and a_k is available pick a_k else pick at random).
- 4) Mark this object as a'_k .
- 5) Remove the last row from the matrix.
- 6) Repeat steps 1-4 until a'_i is defined.

B. Proofs of reasonability

To prove that the sequence provided by the basic algorithm is reasonable we will use the following lemma, which shows that if we have a sequence predicted by the basic algorithm and we remove one object from the set of possible objects, then using the modified algorithm for a player will have an output which will differ from the original output at most at one choice.

Lemma 4.1: Let (a_1, \dots, a_n) be the result of using the basic algorithm on the game G with the set of actions A . By running the modified algorithm for the game G , sequence (a_1, \dots, a_n) and set of objects $A \setminus \{a\}$, where $a \in A$, will give a sequence (a'_1, \dots, a'_n) which will differ from (a_1, \dots, a_n) in at most one element and only if $a \in \{a_1, \dots, a_n\}$.

Case 1: First let us consider the case in which $a \notin \{a_1, \dots, a_n\}$. In this case the resulting sequence will be identical to the original. As in step 4 of the modified algorithm for the k -player the action a_k will still be available and the valuation of objects has not changed, it will be chosen by the k -th player.

Case 2: Now consider the case in which $a \in \{a_1, \dots, a_n\}$. Let us assume that $a = a_k$ for some $1 \leq k \leq n$. Since none of the values have changed and the players from $k + 1$ to n will still have their previous choices available, the algorithm will pick those objects. Of course player k cannot choose a_k because it is not in the set of possible actions. The modified algorithm finds a new object for him which we will mark as a' . It cannot be that $a' = a_i$ for $i > k$ because these objects are already unavailable for the algorithm by now. If $a' \neq a_i$ for all $i < k$ then this will be the only change result of the algorithm, because then all of the objects the modified algorithm has to pick in case of ties are available as in case 1. If $a' = a_i$ for some $i < k$, then it will be assigned to player k , but we have a problem with the assignment for player a_i . Running the modified algorithm for players from $i + 1$ to $k - 1$ will give the same result as the original, by the same reasoning as before. For player i we can repeat the same reasoning as

we did for the player k . As in each such repetition the index will get smaller and the sequence is finite, such replacement will happen only a finite number of times and will result in a sequence which differs only in one element from the original sequence.

Theorem 4.2: Let (a_1, \dots, a_n) be the result of using the algorithm on the game G with the set of objects A . Then (a_1, \dots, a_n) is reasonable.

We will prove this by assigning a move each vertex in the game tree and arguing that we can construct a sub-game perfect equilibrium for the game G for which (a_1, \dots, a_n) is the result. The proof goes by induction on the number of players. The case of $n = 1$ is trivial.

$n = 2$: We have one vertex corresponding to the decision of player 1 from which descent m edges corresponding to all possible actions for player 1. We put a_2 on all vertices of player 2 except the one connected to the edge a_2 . There we can use the modified algorithm on the sequence (a_1, a_2) and the set of objects $A \setminus \{a_2\}$ to find one to put on this vertex. We put a_1 on the root. It should be easy to see that this assignment will produce a sub-game perfect equilibrium with the sequence (a_1, a_2) as a result.

$n - 1 \Rightarrow n$: Now we assume that we can assign moves tree for any game G with $n - 1$ players a given set of objects A which constructs a sub-game perfect equilibrium and a sequence given by the algorithm is the result. We will show how to use this to assign moves in a tree for any game with n players and a sequence (a_1, \dots, a_n) given by the algorithm. We start from a tree for the game G with all vertices empty. For every vertex connected to the root we will run the modified algorithm on the game G without player 1, sequence (a_2, \dots, a_n) , and set of objects $A \setminus \{a\}$, where a is the label of the edge between this vertex and the root. By our inductive assumption, we can construct a full strategy on the subtree starting from that vertex, which has the desired properties and has (a'_2, \dots, a'_n) , given by the modified algorithm, as a result. We can see that by discarding the object a for this whole subtree we can be sure that, as long as we pick the proper object for the root, the whole strategy will stay a perfect subgame equilibrium. What is left is to show that there are no better actions to put at the root than a_1 . Consider first edges a which are not in the set $\{a_2, \dots, a_n\}$. If player one was to pick one of them the result of playing the subtree under that edge, by the lemma, is exactly the sequence (a_2, \dots, a_n) , so it only could be beneficial for him if $v_1^a > v_1^{a_1}$ which is contrary to the way we picked a_1 . Suppose now that player 1 could benefit from committing to an action a from the set $\{a_2, \dots, a_n\}$. By the lemma the resulting sequence (a, a'_2, \dots, a'_n) differs in at most one element from the sequence (a_1, \dots, a_n) . If it differs, then for it to be beneficial it had to be the case that this one action has a greater value for player 1 than a_1 , which is in contrary with the way a_1 was chosen. So there is no action which grants a better result for player 1 than choosing a_1 . We put a_1 on the root getting a proper assignment to the tree which can construct a perfect su-game equilibrium with the result (a_1, \dots, a_n) thus completing the construction.

With multiple preferred objects it could happen that the whole outcome of the game is very different from what the players predicted and in fact the outcome does not have to be a sub-game perfect equilibrium, which would undermine the validity of the reasonable move as a good strategy concept. The next theorem proves that no matter how often the players were wrong in their predictions the whole outcome will be in fact reasonable.

Theorem 4.3: Let G be a game with n players and the set A of available objects. Player 1 uses the basic algorithm to obtain the sequence (a_1^1, \dots, a_n^1) and picks a_1^1 . Then player 2 uses the basic algorithm on the set $A \setminus \{a_1^1\}$, obtains the sequence $(a_1^2, \dots, a_{n-1}^2)$ and commits to a_1^2 . The following players continue in a similar fashion cutting the set of objects. Then the sequence $(a_1^1, a_1^2, \dots, a_1^n)$ is reasonable.

The proof goes by induction on the number of players. The case $n = 1$ is trivial.

$n = 2$: As in the previous proof we have one vertex corresponding to the decision of player 1 from which descent m edges corresponding to all possible actions for player 1. We put a_1^1 on the one vertex of player 1. We put a_1^2 on all vertices of player 2 except the one connected to the edge a_1^2 . We use the modified algorithm for the sequence (a_1^1, a_1^2) and the set of objects $A \setminus \{a_1^2\}$ to find what to place on the last vertex. This strategy will have (a_1^1, a_1^2) as a result. As to show that the assignment can be used to construct a sub-game perfect equilibrium it suffices to notice that even if $a_1^2 \neq a_2^1$ both must be equally valued by player 2 because the algorithm gave those two elements as a possible move of player 2 on two different occasions, while both those actions were available to the player.

$n - 1 \Rightarrow n$: We assume that we can build a strategy tree for any game G with $n - 1$ players and a given set of actions A which is *PSE* and the proper sequence is the result. To show the result for n players we start with a game tree with all vertices empty. For every vertex connected to the root we run the modified algorithm on the game G without player 1, sequence (a_1^2, \dots, a_1^n) and the set of actions $A \setminus \{a_1^1\}$, where a_1^1 is the label of the edge between this vertex and the root. By the inductive assumption the sequence (a_1^2, \dots, a_1^n) is reasonable for the proper sub-game, so the result of the modified algorithm is also reasonable and a sub-game perfect equilibrium can be constructed on this subtree. It remains to argue that after putting a_1^1 in the root the strategy we get a similar result. It is important to notice that the sequence $(a_1^1, a_1^2, \dots, a_1^n)$ is a possible result of using the regular algorithm for the game G with n players and set of objects A . Thus we can use the lemma for all the subtrees. So we can use the exact same argument as in the proof of the previous theorem to show that player 1 cannot benefit from changing committing to another move than a_1^1 .

We can notice that this proof provides more than just the answer to this specific case. First of we didn't explicitly stated if there always exists a pure Nash equilibrium in the case of simultaneous moves by the defenders. We can see that any sub-game perfect equilibrium provided by our algorithm will

remain a Nash equilibrium, if all of the moves are made at the same time and so we see that a pure Nash equilibrium always exists in this case. If we would like to consider a case in which some of the defenders have the resources to defend more than one object we can simulate that by adding copies of that player's valuations to the valuation matrix and dividing one defender with n resources into n players with one resource and identical valuations.

V. SUMMARY

In this paper we have discussed the properties of pure strategies in security games. We recognized which part of the model can be omitted in this situation and which can be redefined to simplify the model. We introduced a situation in which the defenders pick actions in a sequential order, we proved that a pure strategy equilibrium exists in such setting, that we can find a move corresponding to such an equilibrium in polynomial time, and used this construction to prove the existence to a pure Nash equilibrium in the general case of synchronous play of the defenders, as well as when the defenders have more than one resource to use.

As for future directions, the problem discussed in this paper assumed no possibility of communication between the defenders and we think that any query in that direction could be interesting. Also we can see that the result of the game in our case could depend heavily on the order in which the players were able to commit to their strategy. As so, we think that that finding out how much trying to coordinate the players, via an additional player or otherwise, could affect the possible result, or even finding out exactly how many different results can be achieved from a given game in any ordering, could pose also an interesting challenge.

REFERENCES

- [1] J. Nash, "Non-cooperative games," *Annals of mathematics*, pp. 286–295, 1951.
- [2] G. Gottlob, G. Greco, and F. Scarcello, "Pure nash equilibria: hard and easy games," *JAIR*, vol. 24, pp. 357–406, 2005.
- [3] R. W. Rosenthal, "A class of games possessing pure-strategy nash equilibria," *Int. Journal of Game Theory*, vol. 2, no. 1, pp. 65–67, 1973.
- [4] H. Von Stackelberg, *Market structure and equilibrium*. Springer Science & Business Media, 2010.
- [5] J. Pita, M. Jain, J. Marecki, F. Ordóñez, C. Portway, M. Tambe, C. Western, P. Paruchuri, and S. Kraus, "Deployed armor protection: the application of a game theoretic model for security at the los angeles international airport," in *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems: industrial track*. International Foundation for Autonomous Agents and Multiagent Systems, 2008, pp. 125–132.
- [6] J. Tsai, S. Rathi, C. Kiekintveld, F. Ordonez, and M. Tambe, "Iris-a tool for strategic security allocation in transportation networks," *AAMAS (Industry Track)*, pp. 37–44, 2009.
- [7] F. Fang, P. Stone, and M. Tambe, "When security games go green: Designing defender strategies to prevent poaching and illegal fishing," in *IJCAI*, 2015, pp. 2589–2595.
- [8] M. Adrian, "Pure strategies in security games," Ph.D. dissertation, UNICAL Università della Calabria, 2017.
- [9] K. Leyton-Brown and Y. Shoham, "Essentials of game theory: A concise multidisciplinary introduction," *Synthesis Lectures on Artificial Intelligence and Machine Learning*, vol. 2, no. 1, pp. 1–88, 2008.
- [10] S. Tadelis, *Game theory: an introduction*. Princeton university press, 2013.

Formal analysis of timeliness in the RaSTA protocol

Billy Naumann, Christine Jakobs, Matthias Werner

TU Chemnitz

Faculty of Computer Science

Chemnitz, Germany

Email: {billy.naumann,christine.jakobs,matthias.werner}@informatik.tu-chemnitz.de

Abstract—Formal reasoning about the correctness of safety-critical system properties is crucial since such systems may impact their environment when malfunctioning. The Rail Safe Transport Application (RaSTA) Protocol is a protocol for systems used in railway applications such as signaling. It claims to provide highly available and timely communication based on the application’s demands. We investigate timeliness, i.e., the property that application data do not become obsolete.

We analyze the protocol’s specification and provide assumptions necessary to resolve imprecisions. Under the specified error model, we find that the deadlines proposed bound until messages are considered timely is too restrictive, disabling RaSTA’s mechanisms to recover from lost messages in time. We formalize the specification of timeliness to provide a counterexample for the proposed bound and create an improved bound that does not lead to violated deadlines under the same assumptions and error model.

I. INTRODUCTION

THE *Rail Safe Transport Application (RaSTA)* [1] is a protocol used in railway signaling technology between diverse communication endpoints. It is independent of the overlaying application. Since it may be usable for *safety-critical* applications, its correctness is essential. The object of this paper is to formally verify the correctness of a part of the RaSTA protocol. We formally investigate RaSTA’s timeliness property using networks of timed automata and the tool Uppaal for formal reasoning [2].

RaSTA’s specification rests upon natural language. The interpretation of such a specification often relies on either explicit or implicit assumptions, allowing to focus on aspects considered necessary while abstracting from others. Those assumptions pose the danger of creating a model that cannot reflect the wanted properties. The following quote from Sir Tony Hoare shows that formal approaches are a necessary and helpful tool to discuss such assumptions:

The job of formal methods is to elucidate the assumptions upon which formal correctness depends.

In this investigation, we discuss necessary assumptions about imprecisions in RaSTA’s specification, prove that the bound for messages’ timeliness given in RaSTA’s specification is insufficient, and provide an improved bound.

The remainder of this paper is structured as follows: Section II provides an overview of the normative requirements applicable to RaSTA and an overview of the protocol itself, including assumptions made in the RaSTA specification. Section III introduces the formal semantics of networks of timed automata

and Uppaal. In Section IV we present our model and the evaluation of RaSTA’s timeliness property. Finally, section V gives a conclusion of our work.

II. RASTA PROTOCOL

The RaSTA [1] protocol is specified in a pre-standard by the DKE/UK 351.3, a national working committee of the Association for Electrical, Electronic and Information Technologies for railway signaling facilities. It is used in railway signaling technology to achieve safe and highly available communication.

A. Requirements

Strict normative requirements exist, as safety is a crucial concern in this field. RaSTA implements the requirements of [3] for safe communication in open communication systems of category 2, including networks consisting of safety-critical and non-safety-critical systems that can read, write, process, and transmit data. Safety-critical systems use safety-critical transmission functions, assuring Authenticity, Integrity, Timeliness, and Sequence of sent messages. Specifically, RaSTA defines timeliness as a state in which information is available in time according to the requirements. The number of users is generally unknown, as well as their application. Thus unknown amounts of data in arbitrary formats are sent in such networks. There might be routing and management facilities and the communication media may be prone to unforeseen external faults. Authorized access with malicious intentions is explicitly negligible in this category. Thus no cryptographic means are enforced.

The system’s functionality must ensure the aforementioned properties of safety-critical transmission functions. The implementation of such a system implies an evaluation of possible safety threats. Appropriate means must be used to mitigate these threats. In [3], a list of specific safety means is provided, consisting of short descriptions and their requirements.

B. Architecture

RaSTA is implemented between a typical communication stack’s application layer and the transport layer, as pictured in Figure 1. There are only a few requirements on the transport layer, making RaSTA suitable for different scenarios: It must be possible to send messages to specific receivers. The network, including the communication partners, must process the messages in a best-effort manner. It is unnecessary to

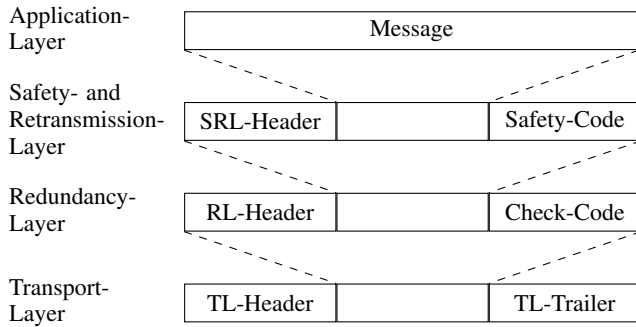


Fig. 1: Architecture of the RaSTA Protocol Stack

TABLE I: Protocol Data Unit of the SRL

Byte(s)	Content
0 - 1	length
2 - 3	type
4 - 7	receiver ID
8 - 11	sender ID
12 - 15	sequence number
16 - 19	confirmed sequence number
20 - 23	timestamp
24 - 27	confirmed timestamp
28 - 28+k-1	payload (k bytes)
28+k - (28, 35, 42)+k	safety code

enforce deadlines, but safety reasons imply fast transmission and adequate quality. Reference [1] states that *commercial of the shelf* transport services such as UDP or TCP are well suited.

RaSTA introduces two new layers between application and transport: the upper *Safety- and Retransmission-Layer (SRL)* and the lower *Redundancy-Layer (RL)*, also shown in Figure 1. The SRL provides a safe communication mechanism for networks according to [3], while the RL aims to provide a highly available communication service via so-called *Redundancy-Channels*. The RL uses multiple transport layer channels (possibly with different transport services) for redundant communication. In this way, messages that get lost or altered on a single transport channel do not affect the communication on SRL-level. Since most of the means to ensure the necessary properties reside in the SRL, this paper only briefly covers the RL.

Table I shows the design of the *Protocol Data Unit (PDU)* of the SRL. We omit a representation of the RL at this point.

The SRL makes use of *IDs* for sender and receiver to ensure authenticity as well as a *safety code* based on the message digest 4 (MD4) [4] algorithm to ensure integrity. Depending on the individual requirements, the latter can take either all, a few, or none of the result's bytes. The protocol uses the *sequence number (SN)* and the *confirmed sequence number (CS)* to maintain the correct sequence of all communicated messages: With each communicated message, *SN* gets incremented. At the same time, *CS* represents the last received *sequence number* of the communication partner. Both *timestamp (TS)* and *confirmed timestamp (CTS)* fields

are used to ensure timeliness. Here, *TS* represents the time at which the sender created the message, and *CTS* represents the last received timestamp of the communication partner, analog to the confirmed sequence number. The insurance of both Sequence and Timeliness requires additional logic, discussed in the protocol specification.

The RL has a more lightweight PDU. There are two primary choices: it uses CRC for its *check code* with different possible configurations. Also, using an additional *sequence number* ensures noticing any race conditions between messages via the individual transport channels. Please note that additional logic is necessary to ensure a correct transmission on the receiver side.

C. Protocol specification

Since RaSTA is a relatively new protocol stack, there is not much work regarding formalizing and verifying its properties, even though its usage in safety-critical scenarios. However, a shortcoming is using MD4 as safety code as described in section II-B, shown by [5]. Here, possible changes to the protocol stack extend RaSTA's abilities to withstand attacks such as the injection of forged messages or replay attacks. Such malicious attacks are not in the scope of RaSTA's requirements for category 2 networks according to [3], but it raises the question if these assumptions are valid in the use case of railway signaling.

RaSTA defines message types for the SRL, used in different situations. Figure 2 shows the abstract state machine for the SRL. Defined are *Connection Request (ConnReq)*, *Connection Response (ConnResp)*, *Disconnect Request (DiscReq)*, *Heartbeat (HB)*, *Data*, *Retransmission Request (RetrReq)*, *Retransmission Response (RetrResp)*, and *Retransmitted Data (RetrData)* messages, from which *Data*, *RetrData* and *HB* messages are defined as relevant for time monitoring. Using the first two message types, connections are established and set up by performing a handshake between both communication partners. This is visualized in Figure 2 with arrows (a), (b), and (c). A *Disconnect Request* message is sent prior to closing an established connection or to indicate errors during the establishment or regular transmission. The corresponding transitions shows Figure 2 as dashed arrows. The communication partners monitor the connection quality via messages of type *HB*. Such Heartbeats are automatically sent after a defined time interval during which no other messages were sent. Application messages carrying a payload are transmitted as *Data* messages. The remaining message types handle error situations: A lost message leads to a corrupted sequence of messages is corrupted. The receiver can recognize this situation, in which he sends a *RetrReq* message. Figure 2 reflects this situation with transition (d). The original sender of the lost or corrupted message answers with a *RetrResp* message (corresponding to transition (f) in Figure 2), after which he repeats all messages with an unconfirmed sequence number as *RetrData* messages. To finalize, a *Data* or *HB* message is sent to indicate that the retransmission is completed, returning to normal operation by transition (e) in Figure

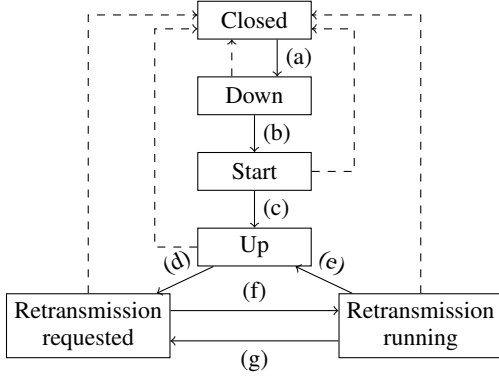


Fig. 2: Abstract State Machine of the SRL [1]

2. Finally, transition (g) corresponds to the situation where another message gets lost during a running retransmission, where another retransmission is initiated.

Besides the abstract states shown in Figure 2, each instance has to manage an internal state, consisting, among other things, of a set of sequence numbers, timestamps, and timers:

- SN_T : the sequence number of the next transmitted message
- SN_R : the expected sequence number of the next received message
- CS_T : the confirmed sequence number of the next transmitted message
- CS_R : the confirmed sequence number of the last received message
- TS_R : the last received timestamp
- CTS_R : the last received confirmed timestamp
- T_{HB} : a timer representing the remaining time until the instance has to send a new message
- T_{DL} : a timer representing the remaining time in which received messages are considered timely

The maximum values for the timer T_{DL} and T_{HB} are defined as configurable parameters, depending on the applications demands. We use $T_{DL,max}$ and $T_{HB,max}$ to represent them.

To transmit data, the sender has to create a new PDU according to Table I using $SN = SN_T$, $CS = CS_T$, $CTS = TS_R$, and $TS = t$, where t is the sender's current timestamp. After creating and sending the message, SN_T is incremented ($SN_T = SN_T + 1$) and T_{HB} is reset to $T_{HB,max}$. The sender has to store a copy of the message until a message with $CS \geq SN$ is received.

The message receiver has to check the sequence of SN , CS , and CTS . First, plausibility checks evaluate if $SN - SN_R$ is below a configured limit. Also, $CS_R \leq CS < SN_T$ has to hold, stating that the received and confirmed sequence number is plausible, too. If these conditions do not hold, the receiver discards the message. Otherwise, he performs more vigorous checks on SN and CTS . A boolean variable $SNinSeq$ is set to true, if $SN_{PDR} = SN_R$ holds. Also, another boolean variable $CTSinSeq$ is set to true, if for messages relevant for time monitoring $0 \leq CTS - CTS_R < T_{DL,max}$ holds. Note that the receiver does not discard the messages if the

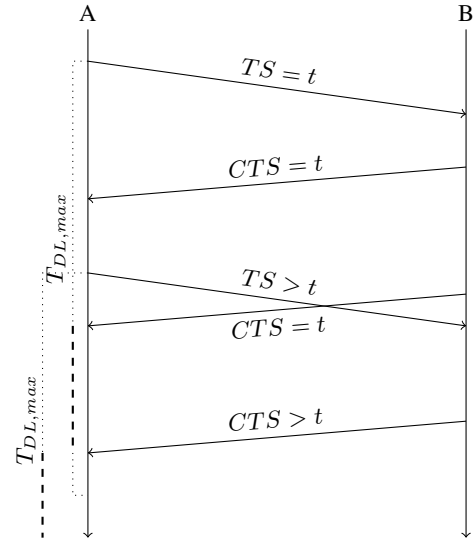


Fig. 3: Adaptive Channel Monitoring

checks result in negative results. The state transitions shown in Figure 2 depend on these variables. Summarizing the resulting behaviors, if $CTSinSeq$ does not hold, the receiver responds by sending a *DiscReq* and closing the connection. If $SNinSeq$ does not hold, the receiver initiates retransmission of all messages with unconfirmed sequence numbers (i.e. $CS_T \leq SN < SN_T$).

D. Timeliness in RaSTA

RaSTA ensures timeliness by applying a concept called *Adaptive Channel Monitoring (ACM)*. The necessary information is included in each SRL-PDU but will be evaluated only for time monitoring relevant messages. According to [1], clocks may not be synchronized and can have different resolutions, consequentially disallowing the interpretation of timestamps of the communication partner.

ACM applies the *Double-Timestamp-Principle*, defined in [3] to check the message round trip times on both communication partners. A sent message carries the sender's local timestamp in TS . Upon receiving this message, the receiver stores this exact value as $TS_R = TS$ in his local state. The following answer of the receiver will carry the confirmed timestamp $CTS = TS_R$ back to the original sender, allowing the round trip time calculation as $RTT = t - CTS$ where t is the current timestamp. A message's timeliness is analyzable by constraining the round trip time since this value overestimates the timestamp of the send-event. Since multiple messages can carry the same CTS value, a round trip completes once a message with a new, greater CTS value arrives. This fact naturally implies that the original sender sent a message with $TS > t$ in the meantime. This message also starts the next round trip, overlapping the current one. This is visualized in Figure 3 for an error-free communication.

After a message with a specific new value of CTS arrives, the receiver updates CTS_R in the local state. All later messages with $CTS = CTS_R$ are considered timely, if

they arrive in an interval of length $T_{DL,max}$ since CTS_R . If another message with $CTS > CTS_R$ arrives, the stored value of CTS_R will be replaced by the new one, indicating that no further messages with the old CTS value should arrive anymore.

To implement these conditions, [1] uses the timer T_{DL} , which is reset with each update to CTS_R to $T_{DL} = T_{DL,max} - RTT$. If this timer reaches its limit, arriving messages may carry outdated information, which would be the case when $CTS = CTS_R$. Figure 3 visualizes this timer as dashed and dotted lines where the dashed portion represents the actual timer running while the dotted portion corresponds to the time since the reference point or until the deadline, respectively.

As $T_{DL,max}$ is a configurable parameter, it is possible to adjust it according to the application's needs. However, [1] states that $T_{DL,max}$ should include enough buffer to take possible retransmissions into account, and gives the following suggestion for its minimal value:

$$T_{DL,max} > 3 \cdot T_{HB,max} + 2 \cdot (T_{AB} + T_{BA}) + T_{RL,seq} \quad (1)$$

Here, according to [1], $T_{HB,max}$ refers to the communication partner's maximum time between two consecutive messages. T_{AB} and T_{BA} indicate the worst-case transmission time of the channel from sender to receiver, where A and B are the communication partners, and $T_{RL,seq}$ indicates the maximum time a message can get delayed in the RL because of race conditions.

As specified in [1], the reasoning behind this formula is as follows: the round trip time of a message can be overestimated by $T_{HB,max} + T_{AB} + T_{BA}$, a lost message will be noticed at worst after $2 \cdot T_{HB,max}$ and the following retransmission can be estimated as $T_{AB} + T_{BA}$, which sums up to the right hand side of Equation 1.

This bound results in an assumed error model where only one message per round trip can be lost. We examine this bound and see it as problematic since it introduces artificial dependencies between the communication partners and their communication channels: As soon as one channel loses one message, the other must deliver correct messages. This assumption is unrealistic since, in reality, the channels themselves cannot share such information.

E. Assumptions regarding RaSTA's specification

Before modeling RaSTA's communication to show timeliness, we need to state some assumptions regarding open or imprecisely defined aspects.

1) *Violation of message sequence*: Since the RL aggregates multiple transport connections between sender and receiver to a single redundancy channel, the correct order of the messages has to be assured since race conditions along the individual channels can occur. The configuration parameter $T_{RL,seq}$ states the delay of messages to be able to restore sequence before delivery to the SRL. We assume that this parameter is set to 0, effectively allowing messages to overtake

each other unhandled. This assumption is reasonable since the SRL notices the incorrect message sequence in the same way a lost message would, triggering the retransmission.

2) *Immediate Responses*: The specification [1] is unclear about internal delays of messages that should be transmitted immediately. This delay is significant in the case of a retransmission of unconfirmed messages. We assume that no additional delay is introduced between two such messages, effectively sending all of them in the correct order simultaneously.

3) *Handshakes*: We focus on analyzing timeliness for the central part of the protocol: The data exchange, in essence, by the behavior of a message's round trip. By that decision, we exclude the handshake to establish the connection and any disconnection semantics. The initial handshake includes a final time-critical heartbeat message. However, this message must be sent immediately after receiving the connection response message. Thus, it will set the initial reference point for the upcoming data exchange. If the receiver discards the message, the handshake fails, and the connection is not established.

Additionally, we reduce the second part of the retransmission handshake, i.e., the *RetrResp*, *RetrData*, *HB* sequence to transmit all missing messages to a single *RetrResp* message. This reduction is feasible since we use the assumptions in Section II-E2, together with the transmission error model. The handshake is only carried out when a communication partner previously discarded a message. Hence, during this handshake, no further messages are discarded. All messages are sent without any delays. Hence, the retransmitting side immediately sends the (final) heartbeat message carrying the $CTS > t$ information. This assumption allows us to abstract from the specific messages to be confirmed.

F. Discussion of RaSTA's timelines property

A derivation of Equation 1 according to [1] is given in section II-D. However, we like to point out that the interpretation of $T_{HB,max}$ as the communication partners parameter is not feasible for all scenarios.

Under the assumption that only one message per round trip can get lost, there are, in essence, two different scenarios that lead to retransmission, shown in Figure 4. Both scenarios share the loss of the information used to finish the round trip, i.e., a timestamp $TS > t$ or a confirmed timestamp $CTS > t$. We describe these scenarios from the view of the final $CTS > t$ message, as the receiver will consider this message's timeliness. Hence, communication partner A is the receiver while B is the sender in the scenarios. In Figure 4a, the receiver's first message containing $TS > t$ is lost during transmission. Subsequently, the sender discards the following message also carrying this information. Since the message sequence is not correct at this point, the sender initiates retransmission. He continues to send heartbeat messages containing the $CTS = t$ information until the finalization of the retransmission. We omitted them for readability reasons. In the worst case, a heartbeat containing $CTS = t$ is sent just before the $TS > t$ information reaches the sender, leading

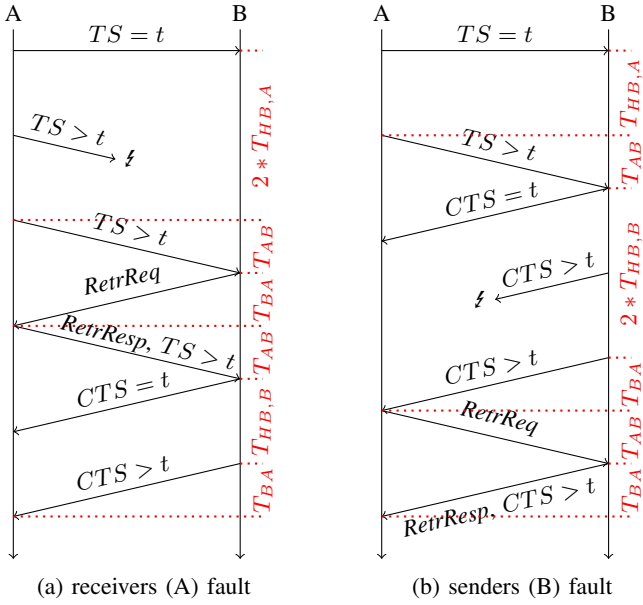


Fig. 4: Retransmission scenarios

to sending the $CTS > t$ information with the next heartbeat. Figure 4a also shows the individual worst case delays.

Accumulated, they lead to the following upper bound shown in Equation 2:

$$T_{DL,A} = 2 \cdot T_{HB,A} + T_{HB,B} + 2 \cdot (T_{AB} + T_{BA}) \quad (2)$$

Analogously, we show the situation where the sender's first message containing $CTS > t$ is lost in Figure 4b. The receiver's message containing $TS > t$ arrives at the sender just after a message containing $CTS = t$ was sent. Hence, the $CTS > t$ is sent as the next heartbeat and gets lost. The receiver discards the next heartbeat since the message sequence is incorrect. At this point, the receiver initiates the retransmission. After this retransmission, the $CTS > t$ information reached the receiver. Equation 3 presents the according upper bound.

$$T_{DL,B} = 2 \cdot T_{HB,B} + T_{HB,A} + 2 \cdot (T_{AB} + T_{BA}) \quad (3)$$

Since both scenarios can occur under our assumptions, we propose to use the maximum of both as the bound for the deadline, referred to by $T'_{DL,max}$, as shown in Equation 4.

$$T'_{DL,max} > \max(T_{DL,A}, T_{DL,B}) \quad (4)$$

Please note that we used $T_{HB,\cdot}$ and $T_{DL,\cdot}$ to indicate $T_{HB,max,\cdot}$ and $T_{DL,max,\cdot}$ for spacing reasons. In the following sections, we show a formal analysis of both limits presented in Equations 1 and 4 using timed automata supported by Uppaal.

III. TIMED AUTOMATA AND UPPAAL

To check the correctness of both limits presented in Equations 1 and 4, we modeled the interesting aspects of RaSTA as timed automata and checked this model with the help of Uppaal.

A. Uppaal

Uppaal [6], [2] is described as "an integrated tool environment for modeling, validation, and verification of real-time systems modeled as networks of timed automata, extended with data types (e.g., bounded integers, arrays)." It is developed jointly by Basic Research in Computer Science at Aalborg University in Denmark and the Department of Information Technology at Uppsala University in Sweden.

Many applications of Uppaal in scientific case studies are shown on Uppaal's website [2], such as the verification of different versions of the well-known Fischer Protocol [7] for mutual exclusion in [8]. Nevertheless, also industrial protocols such as the Philips Audio Protocol for exchange of control information [9], or the Bang and Olufsen Audio/Video Protocol for transmission of messages between audio and video components over a single bus [10] have been model checked by Uppaal. Primarily the latter was known to be faulty. Uppaal's generation of (erroneous) traces allowed us to find the error. Also, Uppaal was used to find and verify a fix for this problem.

B. Modelling

Timed automata exist in multiple flavors, but they generally combine the known concept of finite state machines and clocks, unique variables used to represent time. Uppaal uses a dense time model, where clocks evaluate real numbers and advance synchronously. For evaluation, Uppaal can express clock valuations as symbolic constraints, thus reducing the state space by collapsing all clock valuations that share common properties. Further, Uppaal allows systems to be modeled as networks of timed automata by composition from individual automata. Every automaton may engage in (enabled) transitions, also used to synchronize multiple automata. [11]

In [11], the definition of the Timed Automata used in Uppaal is as follows:

A timed automaton \mathcal{A} is a tuple (L, l_0, C, A, E, I) , where L is a set of locations, l_0 is the initial location, C is the set of clocks, A is a set of actions, co-actions and the internal τ -action, $E \subseteq L \times A \times B(C) \times 2^C \times L$ is a set of edges between locations with an action, a guard and a set of clocks to be reset, and $I : L \rightarrow B(C)$ assigns invariants to locations.

Reference [11] also provides the definition and semantics of a network of timed automata, consisting of n timed automata $\mathcal{A}_i, 1 \leq i \leq n$. The location vector $\vec{l} = (l_1, \dots, l_n)$ corresponds to the locations of each individual automation. Further, the invariants are merged to an invariant function over the location vectors $I(\vec{l}) = \wedge_i I_i(l_i)$. Finally, the notation $\vec{l}[l'_i/l_i]$ denotes the location vector where the i th element l_i is replaced by l'_i .

The semantics are then given by [11] as follows: Let $\mathcal{A}_i = (L_i, l_i^0, C, A, E_i, I_i)$ be a network of timed automata and $\vec{l}_0 = (l_1^0, \dots, l_n^0)$ the vector of initial locations. The semantics is defined as a labelled transition system $\langle S, s_0, \rightarrow \rangle$ with $S \subseteq (L_1 \times \dots \times L_n) \times \mathcal{R}^C$ as the set of states, $s_0 = (\vec{l}_0, u_0)$ is the initial state and $\rightarrow \subseteq S \times S$ is the transition relation such that:

- $(\bar{l}, u) \xrightarrow{d} (\bar{l}, u+d)$ if $\forall d' : 0 \leq d' \leq d \implies u+d' \in I(\bar{l})$, and
- $(\bar{l}, u) \xrightarrow{a} (\bar{l}[l'_i/l_i], u')$ if $\exists l_i \xrightarrow{\tau g r} l'_i : u \in g, u' = [r \mapsto 0]u$ and $u' \in I(\bar{l}[l'_i/l_i])$.
- Further, $(\bar{l}, u) \xrightarrow{a} (\bar{l}[l'_j/l_j, l'_i/l_i], u')$ if $\exists l_i \xrightarrow{c?g_i r_i} l'_i \wedge l_j \xrightarrow{c!g_j r_j} l'_j : u \in (g_i \wedge g_j), u' = [r_i \cup r_j \mapsto 0]u \wedge u' \in I(\bar{l}[l'_j/l_j, l'_i/l_i])$

Hence, possible transitions are categorized in *delay* and *action* transitions. The former are described by $(\bar{l}, u) \xrightarrow{d} (\bar{l}, u+d)$, letting the system evolve in time for d time units by mapping each clock $c \in C$ to the value $u(c) + d$, if the invariants of all locations aren't violated for any time point until the delay has occurred. The latter corresponds to a single edge or a pair of edges in the automaton. Here, either a single automaton or a pair of automata in the network change their locations according to action a in $(\bar{l}, u) \xrightarrow{a} (\bar{l}[l'_i/l_i], u')$ and $(\bar{l}, u) \xrightarrow{a} (\bar{l}[l'_j/l_j, l'_i/l_i], u')$ respectively. Either are only possible if their guards are satisfied. Taking the transition resets all referenced clocks, and the resulting locations' invariants must be satisfied. When two automata perform such a pairwise transition, it is also necessary to label the transitions by corresponding co-actions, expressing active and passive synchronization at these points.

Uppaal extends such networks of timed automata with many features shown in the following list, as well as other constructs borrowed from C-like programming languages such as arrays, initializers, record- and custom-types as functions. [11]

- *Templates* for instantiating automata
- Invariants over *internal state variables*
- *Non-deterministic choice* of binary synchronization channels when multiple co-actions are possible
- *Urgent and committed locations* which disallow the passage of time

Within such templates, Uppaal uses additional labels for locations and edges, which allow to define the behavior of the automaton in an easy way, for example, to express a location's invariants. Actions can have *select* labels, which can be used to non-deterministically bind values from a given range to variables which can then be used in the remaining labels of the action. Also, *guards* enable actions upon fulfillment or disable them in case of violation. *Synchronization* labels allow the use of synchronization channels, where edges with complementary synchronization labels $c!$ and $c?$ over a shared channel c synchronize on taking the $c!$ labeled action, reassembling co-actions. Finally, *update* labels can alter the current internal state by changing variables' values or assigning values to clocks. [11], [12]

Figure 5 shows an example of a timely bounded synchronous communication channel. We use this channel to model the communication between two RaSTA communication partners. Messages are accepted from the sender via the `send?` co-action, transiting from the `Idle` to the `Transmitting` location. The variable `content` refers to the channel's content, taken from whatever resides in `data_send`, the senders send buffer. To reduce the state space, the send buffer is

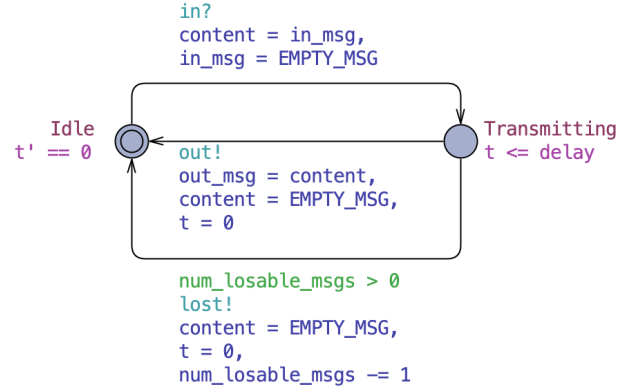


Fig. 5: Model of the RaSTA Communication Channel

reset to empty as soon as the channel has accepted the message. As long as $t \leq \text{delay}$, the channel may reside in the `Transmitting` location, transitions back to `Idle` are possible immediately. There are two possible transitions corresponding to the correct and faulty transmission. In the latter case, the channel's content is not altered and copied into the receive buffer. Instead, the `lost!` co-action indicates an altered message to discard. The clock t is reset when transiting back to `Idle`, where it is stopped via the location's invariant. These measures are also made to reduce the state space.

To define a model, Uppaal uses a *system definition*, which allows instantiating templates to *processes*. Here, it is possible to bind the template's parameters to actual values or define partial instantiations to reuse similar processes. They are composed concurrently to a system by enumerating them after the `system` keyword.

C. Verification

Uppaal allows checking different properties for a system via model checking. Queries to the model express these properties using a simplified version of *Timed Computation Tree Logic (TCTL)* [13]. Queries consist of the path- and state formulae, where state formulae describe individual states and path formulae quantify over the model's traces. Uppaal does not allow the nesting of path formulae [11]. Such formulae are categorized depending on their semantic and matched runs. We explain the used state formulae and safety formulae in the following sections. Additionally, the standard version of Uppaal supports reachability formulae and liveness formulae. Note that many extensions extend classical Uppaal, for example, by examining statistical properties.

1) *State Formulae*: State formulae express the properties of individual states without considering the model's behavior. They are similar to guards in that they are described by side-effect-free expressions, for instance, $x == 42$. Besides statements over internal variables, it is also possible to test if a automaton A is in a certain location l by the expression $A.l$. Internal state variables of a single Automaton are accessible in the same way. Further, deadlocked states (where no outgoing action transitions from the state or delayed successors are possible) can be expressed via the keyword `deadlock`. [11]

2) *Safety Formulae*: Safety Formulae describe that *something bad will never happen*. A general technique is to express something bad in the model's terms, for instance, the violation of a deadline, and then invariantly assure that the model never fulfills this condition. Analogously, the model must fulfill *something good* invariantly. Uppaal uses TCTL formulae $A\Box\phi$, which are expressed in Uppaal as $A[\]\ \phi$ for a state formula ϕ , expressing that ϕ must be true for all reachable states. [11]

IV. VERIFYING TIMELINESS

While it is possible to build a system in Uppaal reassembling the whole SRL and find a suitable formula to describe timeliness in this model, we decided to abstract from this approach for multiple reasons. First, we are only interested in the property of timelines. Therefore, it is feasible to abstract from unnecessary parts of the protocol. Our model abstracts the exact Protocol Data Unit shown in Table I while keeping each message's sequence and time information. Note that it is possible to omit *CS* since the receiver will discard messages if the plausibility checks fail as described in section II-C. Also, by maintaining an appropriate communication channel model, one can abstract from the RL, leaving only a concentrated portion of the protocol for verification. It is possible to reduce this model further since we are only interested in the timeliness of a single message, expressed as a round trip as shown in Figure 3. This reduction is possible since RaSTA claims to ensure timeliness for all time-critical messages, so it is sufficient to find a single situation where RaSTA fails to do so to show the violation of this property. This reduction aims to find a Uppaal model with only a few locations and internal state variables.

Most model checking tools try to explore the whole state space of the model by finding all possible execution paths and states on them. This approach becomes infeasible quickly since the number of possible states grows exponentially with the number of used internal state variables. Ultimately, this *state space explosion* leads to an infeasible time demand for evaluating the properties. There exist ways to approach this problem, such as symmetry reductions, but especially for software verification, this problem is still not solved [14]. This issue emphasizes the importance of a compact and abstract model.

A. Modelling assumptions

We discussed some possible abstractions and why they are feasible at the beginning of this section. Such abstractions' bases are usually on assumptions that restrict the system's modeled behavior in a certain way. While such abstractions allow the formulation of a simpler model, it is essential to ensure that the result is still a valid model of reality, including all necessary aspects of the system to reason about the properties of interest. Otherwise, the model might still be correct but becomes irrelevant since it does not lead to any desired statements.

We have shown assumptions necessary to formalize the specification of RaSTA in Section II-E. The following sections discuss the consequences of formalization and make assumptions regarding which aspects the model of the protocol stack needs and which can be abstracted.

1) *Redundancy Channels*: As described in section II-E1, we assume that the redundancy channels used in the RL lead to possible violations of message sequence. Since we are not interested in showing availability improvements, we decided to abstract from the RL and model only the SRL. The underlying communication channels are seen as per message, meaning that a new virtual communication channel is available for each message. Since Uppaal supports the dynamic instantiation of templates only for statistical queries, we have to limit ourselves to a constant number of available channels, thus limiting the number of messages sent simultaneously. Since, in real-world scenarios, communication always is limited by a specific throughput, we find this assumption feasible. We restrict our analysis to the case where $T_{AB} < T_{HB,max,A}$ and analogously for the values of B. This restriction allows the assumption of FIFO channels so messages cannot overtake each other. Alternating messages during transmission, leading to discarding the message by its receiver, is still possible. Such errors have the same impact on possible retransmissions initiated by the SRL.

2) *Message semantics*: Heartbeat and Data messages share the same information except for an empty application payload. The sender sends Heartbeat messages only when the application is ready to send (*Data*) messages in a defined time interval since the last message. Since RaSTA is independent of the overlying application, we can use this fact to abstract from both message types and reduce them to a single kind of message.

3) *Timestamp relationship*: Since we aim to show timeliness for a single message round trip, we can abstract from the specific values of the timestamps and use a relative representation during this round trip. Additionally, such a relative representation can abstract from the actual values since only the relationship between their corresponding send- and receive-events is necessary to capture the behavior, as shown in Figure 3. Hence, we directly represent the relationship between the ongoing time and the messages *TS* and *CTS* values using state variables.

4) *Message Sequence*: Another critical assumption is that the specific values of the sequence numbers and confirmed sequence numbers do not matter to show the timeliness of a single message. The sequence numbers are used to trigger retransmission if *SNinSeq* is false, as described in Section II-C. The receiver sets this flag if two consecutive messages do not have consecutive sequence numbers. To be able to abstract from the specific values, we inform the receiver of a message about the alternation of the message. In reality, the receiver would check the message's integrity via the safety code and discard it if the check fails. Hence, we update *SNinSeq* before the reception of the following message. The receiver can then react appropriately based on this information by

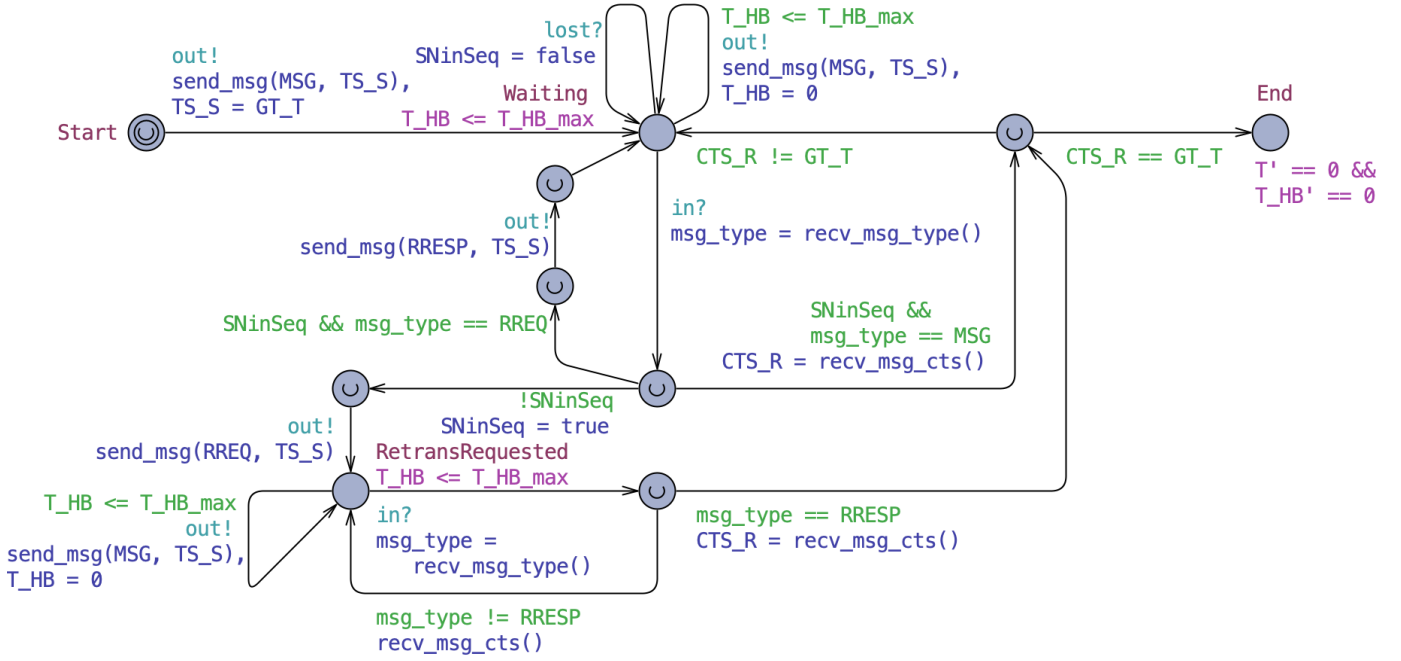


Fig. 6: Model of the RaSTA Receiver

engaging in a retransmission handshake.

5) *Immediate Responses:* We assume that the sender will not delay messages from being sent immediately. There are two significant points in the protocol where this is important: The initial *ConnReq*, *ConnResp*, *HB* handshake to establish the connection and the handshake to perform retransmission, including the transmission of the *RetrData* messages carrying the previously lost messages. As stated in II-E2, [1] covers this not explicitly: The visualizations show a delay of one time unit per message while the descriptions state that the sender has to send messages immediately.

6) *Number of retransmitted messages:* When a message is lost, the sender of this message must retransmit all unconfirmed messages, as described in Section II-C. Such messages may be sent before the modeled round trip. Hence, it is impossible in our model to find a representation of the messages to be retransmitted. However, the only information about these messages we care about is if one of them will complete the current round trip. Especially the final *HB* message will contain the $CTS > t$ information. Therefore, we reduced the set of retransmitted messages to a single one carrying the relevant information. This reduction is feasible since we already assumed that there is no additional delay between immediately sent messages in Section IV-A5.

B. Model and Verification

In this section, we show our model and the analysis of the SRL of RaSTA in Uppaal based on the assumptions shown in the previous sections. Since we want to model a single round trip, our model uses asymmetrical behavior, even if the RaSTA protocol is symmetrical after connection establishment. Hence, individual templates in Uppaal model sender and receiver.

The channel is also an individual template model, which is instantiable multiple times to enable communication between sender and receiver.

Since we abstract from the specific values of the timestamps, it is necessary to model the relation between the reference point t^* when the receiver sends an initial message and the values of TS_R and CTS_R . We defined constant values EQ_T and GT_T to express if a received (confirmed) timestamp is equal to or greater than t^* . Transmitted messages carry this information instead of concrete time information.

Both sender and receiver use a clock T_{HB} . This clock represents the time since sending the last heartbeat message. With appropriate location invariants and action guards, we enforce that both communication partners never send two consecutive messages more than T_{HB_max} time units apart, where the concrete value depends on the chosen parameter for sender and receiver. Additionally, we use a clock T for the receiver model to indicate the current time during the round trip, by which the receiver determines the duration until the arrival in the *End* location, where this clock is stopped.

Further, sender and receiver use urgent states whenever a message is received and during retransmissions. This reflects that the time for the evaluation of the messages header and the decision of the upcoming actions is negligible and serves as an implementation of assumption in section II-E2.

The upcoming sections describe the individual templates.

1) *Receiver:* The Receiver's model is shown in Figure 6. As the evaluation of a message's round trip time depends on a defined reference timestamp, the receiver starts sending a message carrying its current timestamp t^* in TS , which is implemented in the `send_msg(MSG, TS_S)` update. At this point TS_S has the value EQ_T and is updated to GT_T

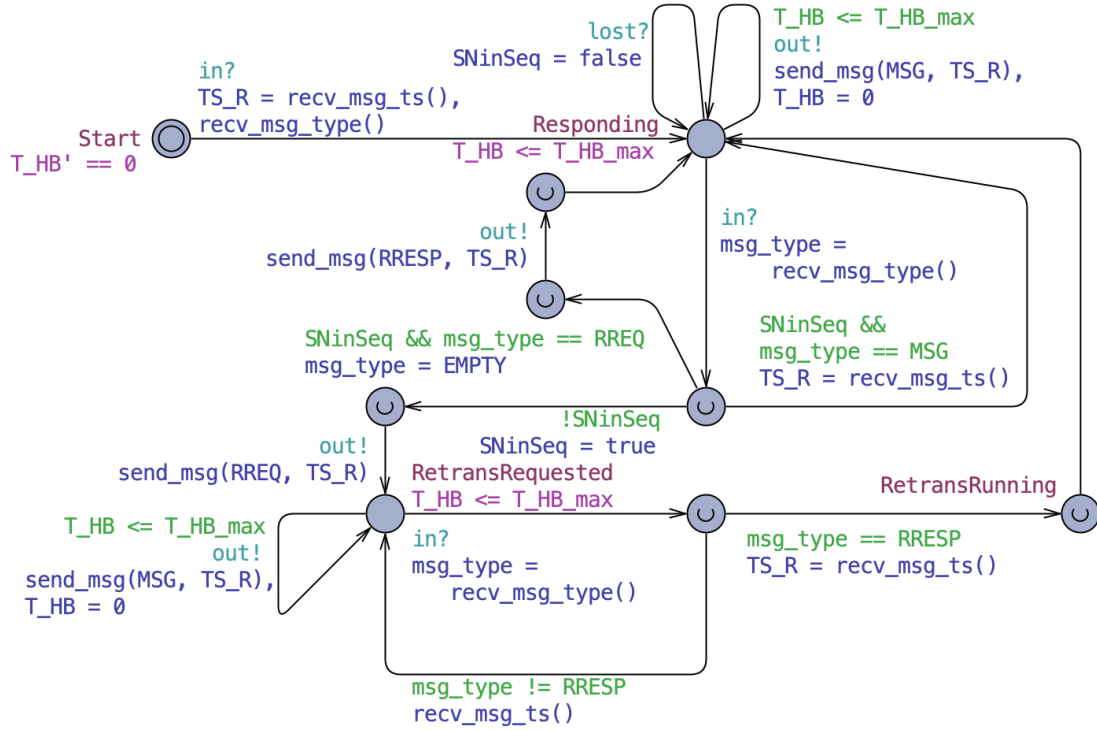


Fig. 7: Model of the RaSTA Sender

after the message is sent. The receiver will use this information for sending all further messages. He will reside in the location *Waiting* until either a new message has to be sent or a message is received or indicated as lost. By the assumption in section IV-A2, we abstract from distinct *HB* and *Data* messages at this point. We do not abstract from *RetrReq* messages. Such messages are identified via the *msg_type* variable and handled appropriately. The clock T_{HB} together with the location's invariant and the transition guards constrain the time between sending such messages. When a message is indicated as lost, the receiver sets $SNinSeq = false$. In reality, this would happen later when the following message is received, but since this is causally dependent on a lost message in our model, we decided to set this flag at this point. When a message is received, the receiver checks $SNinSeq$ and either updates its CTS_R value based on the information in the message or triggers retransmission. If the message is a *RetrReq*, indicated by $msg_type == RREQ$, the according *RetrResp* message is sent immediately, carrying GT_T as its timestamp. By assumption in Section II-E3, this is the only message necessary to complete this retransmission in our model. Otherwise, and when no retransmission is necessary, the receiver either returns to the *Waiting* location when the received CTS value is less or equal to t^* , i.e., $CTS_R != GT_T$. In the same case, he moves to the *End* location, indicating that a new reference timestamp was confirmed by the sender via $CTS_R == GT_T$, thus ending the round trip. If retransmission is necessary, the receiver will initiate the corresponding handshake by sending the *RREQ* message

and transitioning to the *RetransRequested* location. Here the receiver continues to send messages as in the *Waiting* location and waits until the retransmitted messages arrive. All other messages with $msg_type != RRESP$ are ignored, their information will be part of the *RetrResp* message. Upon receiving his message, the receiver evaluates the contained CTS_R as for regular messages.

2) *Sender*: In our model, the sender takes the role of sending the messages whose timeliness is subject to verification. The sender starts by receiving the reference point t^* as its first message, represented by the value of TS_R , which is EQ_T at this point. At this point, the sender transitions into the *Responding* location. From here on, the model is similar to the receiver, differing only in interpreting the message's timing information. Where the receiver uses this information in the CTS_R variable, the sender uses it to update the state of the TS_R variable. Also, the *End* location is absent since the sender is not informed about finishing the round trip.

3) *Channel*: As described in Section IV-A1, we model the *RL* and underlying channels as per message. Uppaal supports the dynamic instantiation of templates only for statistical queries. Hence we are forced to limit our model by over-approximating the number of sent messages. At this point, a single channel will act synchronously, only accepting messages for transmission when it is in *Idle* and only allowing the delivery returning from the *Transmitting* location, as shown previously in Figure 5. As long as the channel is in location *Idle*, its clock t is stopped at 0 by the location's invariant. When a message is accepted via the *in?* co-action, the channel transitions to the *Transmitting* location, and

after at most the worst-case transmission delay delay the message is either delivered or lost. Both cases are modeled as a transition back to the `Idle` state, resetting both the internal state of the channel and its local clock t . Only for successful transmission, the channel copies the message to the receiver's buffer `out_msg`, the receiver is synchronized here via the `out!` co-action. The co-action `lost!` is used for a faulty transmission, where the channel ignores the message's contents. In the case of an erroneous transmission, the channel also decrements the variable `num_losable_msgs` to 0 to indicate that no further messages should be lost.

4) *Verification and Results:* Since, in our model, the absence of missed deadlines describes timeliness, the formulation of the property is possible as a safety formula, stating that there is no case of deadline violation. We checked both the original value shown in Equation 1 as well our adapted deadline shown in Equation 4 via the following formulae for verification in Uppaal, where both the values for $T_{DL,max}$ and $T'_{DL,max}$ are calculated as the bounds based on the parameters for the heartbeat and worst-case transmission times in Uppaal.

$$A\Box(\text{Receiver.End} \implies \text{Receiver.T} \leq T_{DL,max}) \quad (5)$$

$$A\Box(\text{Receiver.End} \implies \text{Receiver.T} \leq T'_{DL,max}) \quad (6)$$

Since Uppaal doesn't allow symbolic constants for model parameters, we used the values $T_{HB,max,A} = 5$, $T_{HB,max,B} = 3$, and $T_{AB} = T_{BA} = 1$, resulting in the bounds $T_{DL,max} = 13$ and $T'_{DL,max} = 17$. With these values, we could show that the proposed bound $T_{DL,max}$ by [1] is violated while our bound is still satisfied.

Even though we aim to instantiate channels for each message individually, the increased state space limits the feasibility of the evaluation. We decided to restrict the channel model to FIFO channels by instantiating only one pair of channels between sender and receiver. As shown in section IV-A1, this limits the validity of our results to the case where $T_{AB} < T_{HB,max,A}$, as in this case, heartbeats are not affected by unavailable channels.

We were able to show for a few selected values that this property holds, but a general statement for all possible assignments is not possible in this way. However, our model can be used to verify the timeliness of a concrete RaSTA communication instance within our assumptions.

V. CONCLUSION AND FUTURE WORK

Safety-critical systems, such as railroad communication networks, demand a clear and comprehensible analysis of all aspects that potentially affect the correctness and safety of the user and the environment. Our analysis shows where the specification of RaSTA is unclear regarding timeliness. We were able to show that the recommended deadline for the RaSTA communication protocol is not guaranteed to hold for the corresponding error scenario. This inherent violation

demonstrates that using formal methods for software verification is a viable approach not only to show formal correctness where necessary but also to elucidate underlying assumptions.

While we were able to show that the proposed bound is not sufficient, we could not provide a complete formal verification of the correctness of our bound. This open end is caused primarily by inherent problems of model checking, e.g., the state space explosion when stepping back from specific abstractions, such as using concrete timestamps instead of our approach. We aim to encounter the use of more general communication channels by lifting our FIFO assumption. Also, we will deal with more complex scenarios, for example, where the worst-case transmission delay is higher than the deadline for sending heartbeats.

REFERENCES

- [1] "Electric signalling systems for railways - part 200: Safe transmission protocol according to DIN EN 50159 (VDE 0831-159)," Jun. 2015.
- [2] Home - Uppaal. Date accessed: 2022-21-04. [Online]. Available: <https://uppaal.org>
- [3] "Railway applications - communication, signalling and processing systems - safety-related communication in transmission systems; german version EN 50159:2010," Apr. 2011.
- [4] R. L. Rivest, "The MD4 Message-Digest Algorithm," Internet Requests for Comments, April 1992, <http://www.rfc-editor.org/rfc/rfc1320.txt>. [Online]. Available: <http://www.rfc-editor.org/rfc/rfc1320.txt>
- [5] M. Heinrich, J. Vieten, T. Arul, and S. Katzenbeisser, "Security analysis of the rasta safety protocol," in *2018 IEEE International Conference on Intelligence and Security Informatics (ISI)*, 2018. doi: 10.1109/ISI.2018.8587371 pp. 199–204.
- [6] J. Bengtsson, K. G. Larsen, F. Larsson, P. Pettersson, and W. Yi, "Uppaal—a tool suite for automatic verification of Real-Time Systems," *BRICS Report Series*, vol. 3, no. 58, Jun. 1996. doi: 10.7146/brics.v3i58.18769. [Online]. Available: <https://tidsskrift.dk/brics/article/view/18769>
- [7] L. Lamport, "A fast mutual exclusion algorithm," *ACM Trans. Comput. Syst.*, vol. 5, no. 1, p. 1–11, jan 1987. doi: 10.1145/7351.7352. [Online]. Available: <https://doi.org/10.1145/7351.7352>
- [8] K. G. Larsen, P. Pettersson, and W. Yi, "Compositional and Symbolic Model-Checking of Real-Time Systems," in *Proc. of the 16th IEEE Real-Time Systems Symposium*. IEEE Computer Society Press, Dec. 1995. doi: 10.1109/REAL.1995.495198 pp. 76–87.
- [9] —, "Diagnostic model-checking for real-time systems," in *Hybrid Systems III*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1996. doi: 10.1007/BFb0020977. ISBN 978-3-540-68334-6 pp. 575–586.
- [10] K. Havelund, A. Skou, K. G. Larsen, and K. Lund, "Formal modeling and analysis of an audio/video protocol: An industrial case study using uppaal," in *Proceedings Real-Time Systems Symposium*. IEEE, 1997. doi: 10.1109/REAL.1997.641264 pp. 2–13.
- [11] G. Behrmann, A. David, and K. G. Larsen, *A Tutorial on Uppaal*, ser. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, vol. 3185, p. 200–236. ISBN 978-3-540-23068-7. [Online]. Available: http://link.springer.com/10.1007/978-3-540-30080-9_7
- [12] Uppaal documentation. Date accessed: 2022-21-04. [Online]. Available: <https://docs.uppaal.org/>
- [13] T. Henzinger, X. Nicollin, J. Sifakis, and S. Yovine, "Symbolic model checking for real-time systems," *Information and Computation*, vol. 111, no. 2, p. 193–244, Jun 1994. doi: 10.1006/inco.1994.1045
- [14] E. M. Clarke, W. Klieber, M. Nováček, and P. Zuliani, *Model Checking and the State Explosion Problem*, ser. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, vol. 7682, p. 1–30. ISBN 978-3-642-35745-9. [Online]. Available: http://link.springer.com/10.1007/978-3-642-35746-6_1

Channel-Less Process Communication

Tomas Plachetka

Comenius University, Bratislava

Faculty of Mathematics, Physics and Informatics

Email: plachetka@fmph.uniba.sk

Abstract—A channel is an abstract data structure which allows for passing messages from one process to another one. We propose several variants of OCCAM, a minimalistic programming language in which a program consists only of processes and channels. The variants differ in how channels are accessed by processes. We prove that all these variants are equally expressive, i.e. an arbitrary OCCAM program can be simulated in any of the variants and the other way around. A particularly interesting variant is to assign exactly one channel to each parallel process. This makes the concept of channels redundant, provided that the parallel processes are named. The simulation techniques can be applied to a variety of abstract models and practical systems.

I. INTRODUCTION

CHANNELS are widely used in abstract models of communicating parallel processes [1], [2], [3], [4], as well as in computer programming languages [5], [6], [7] [8], [9], [10], and hardware descriptions [11], [12], [13]. In operating systems, Unix pipes [14] directly correspond to channels.

The main contribution of this paper is showing a transformation of a channel-based programming language to an equally expressive programming language in which channels and processes become a single entity. We illustrate this transformation on OCCAM [5], [15], which is based on the synchronous abstract model CSP (Concurrent Sequential Processes) [1] and belongs to practical programming languages whose semantics has been formally defined [16], [17], [18], [19], [20]. Although OCCAM is “pure and small”, a sharper Ockham’s razor trims it even more. Along with channels, we also eliminate nesting of parallel processes and the ALT constructor from OCCAM.

The motivation of this work is not solely theoretical. Many programming languages build on a message passing paradigm without channels, e.g. MPI [21], Erlang [22] and Akka (JVM) [23]. An important question is whether the absence of channels is restraining. This paper suggests a negative answer. A consequence is that channel-based (OCCAM-like) programming languages are intrinsically redundant.

The paper is organised as follows. Section II presents a relevant subset of the OCCAM language (leaving out unnecessary details, occasionally using an abbreviated syntax). In Section III, it is shown that nesting of parallel processes can be replaced by a flat process structure (a variant OCCAM-1PAR). In Section IV, the directional graph interconnection of processes and channels is replaced with a hypergraph

interconnection which uses shared channels (OCCAM-SH). In Section V, a concrete hypergraph structure is proposed which leads to a unification of process and channel identifiers, i.e. to a channel-less model (OCCAM-CL). All the variants OCCAM, OCCAM-1PAR, OCCAM-SH, and OCCAM-CL are equally expressive. Section VI concludes the paper.

II. OCCAM

OCCAM was targeted to Transputers [24], single-chip computers specifically designed to support parallel programming. Transputers could be easily connected to form a network, process scheduling and communication were implemented in the hardware. Although Transputers were discontinued in 1990’s, they could in some parameters compete with contemporary computers (e.g. context switch below 1 μ sec still belongs to the fastest ever). OCCAM found its followers, e.g. OCCAM- π [7] and Rain [8].

An OCCAM program consists of a finite number of processes and a finite number of channels; these numbers are known before the program starts and do not change in runtime. A process in OCCAM is either an atomic process ($:=$, assignment; $?$, input from a channel; $!$, output to a channel; SKIP, which does nothing and terminates), or a compound process. Constructors of compound processes (SEQ, PAR, IF, WHILE, ALT) combine processes into a single one. Processes in the SEQ constructor are executed sequentially in the given order, i.e. when a process terminates, the following one becomes active. SEQ terminates when its last mentioned process terminates. The constructors IF, WHILE and the assignment process behave in a common way, except for IF, in which the conditions are always tested in the order they are written, and (only) the process under the first satisfied condition becomes active (IF terminates when this process terminates).

The replicator FOR can be used with constructors. Replication works as a macro expansion. For example, SEQ $i=0$ FOR 2 p expands to SEQ p p , with $i = 0$ in the first replica of the process p , and $i = 1$ in the second one. This corresponds to a sequential repetition of the process p in a for-loop.

The scope of a variable is limited to the process following the variable’s declaration (e.g. INT v :), including processes nested in that process (variables used in replicators are not declared). Usual primitive types are available. The only compound type is an array, indexed from 0.

Processes of the PAR constructor run in parallel and have read-only access to variables which are shared in their scopes. A process is allowed to change (using $:=$ or $?$) only the

This research has been supported by the grant 1/0601/20 of the Slovak Scientific Grant Agency VEGA.

variables which it does not share with another parallel process. The only way how parallel processes may influence one another is via channels which are declared as variables of type `CHAN`. `PAR` terminates when all its processes terminate.

Channels in OCCAM are unidirectional and unbuffered. Each channel connects exactly two parallel processes—one reads from the channel, the other one writes to the channel. Parallel processes can be depicted as vertices and channels as edges of a directed graph (an edge points from the writer to the reader). Reading from a channel (`?`) blocks until a `!` process writes to the channel; and conversely, writing to a channel (`!`) blocks until a `?` process reads from the channel). The corresponding `?` and `!` both terminate after the message has been transferred across the channel from the `!` process to the `?` process. A message is a finite sequence of values (of primitive types or arrays).¹

The `ALT` constructor multiplexes reading from several input channels.² While reading from each single input channel would block, the `ALT` constructor blocks. When reading at least one input channel would terminate (we say that such a channel is readable), a message is read from one readable channel and a process in the corresponding branch takes over. When this process terminates, then `ALT` terminates.

The choice of the readable channel inside the `ALT` constructs and the scheduling of the parallel processes is not under program's control. The scheduler executing the program is unpredictable (nondeterministic) in making these choices. This does not mean that it has to be "random". For example, it is allowed to (but does not have to) prioritise the readable channels in the order they are mentioned in `ALT` constructs. Similarly, it is allowed to (but does not have to) prioritise the execution of active parallel processes in the order they are mentioned in `PAR` constructs. A correct program must guarantee its intended behaviour for all possible schedules.

Processes and channels are static, i.e. they are constructed before the execution of a program. During the execution, each process is in one of the following states: active, passive, blocked. The program evolves according to the rules described above (see [18] for details). At the beginning of the execution of a program, the first process is active, all other processes are passive. A termination of an active process does not mean that the process ceases to exist—it just changes its state to passive. The program execution terminates when there are no active processes. (We can distinguish between a "correct termination" where all processes are inactive, and a deadlock where at least one process is blocked.)

¹Although OCCAM requires a declaration of types of messages which are passed over channels (so-called protocols, e.g. `CHAN OF INT; BOOL ch;`, we consistently use type-less channel declarations throughout the paper. These correspond to `CHAN OF ANY` declarations in OCCAM, where the programmer is responsible for ensuring that the sequences of values in messages transferred over a channel from a `!` process are of the same types as the corresponding variables in a `?` process (so that an incoming message can be stored to the variables in the `?` process).

²For the sake of simplicity, we do not consider `ALT` with boolean expressions attached to the channel inputs (so-called guarded `ALT`).

Fig. 1 shows two OCCAM programs used as running examples throughout the paper. The programs simulate each other. For an arbitrary schedule, both programs terminate and the variable m attains values 0, 1, 2, 3 in one of the following orderings: [0, 1, 2, 3], [0, 1, 3, 2], [1, 0, 2, 3], [1, 0, 3, 2]. The actually observed ordering depends on a concrete schedule. For an arbitrary schedule of P_1 , a schedule of P_2 exists such that the orderings match, and vice versa. Moreover, there is a bijective correspondence of processes which run in parallel in P_1 and P_2 , regardless of their nesting in `PAR` constructs (we refer to line numbers in P_1 and P_2): [5, 5], [11, 12], [12, 13]. In this sense, P_1 and P_2 are equivalent.

Program P_1	Program P_2
1: CHAN ch1, ch2:	1: CHAN ch1, ch2:
2: SEQ i = 0 FOR 2	2: SEQ i = 0 FOR 2
3: PAR	3: PAR
4: INT m:	4: INT m:
5: SEQ j = 0 FOR 2	5: SEQ j = 0 FOR 2
6: ALT	6: ALT
7: ch1 ? m	7: ch1 ? m
8: SKIP	8: SKIP
9: ch2 ? m	9: ch2 ? m
10: SKIP	10: SKIP
11: ch1 ! 2 * i	11: PAR
12: ch2 ! (2 * i) + 1	12: ch1 ! 2 * i
	13: ch2 ! (2 * i) + 1

Fig. 1. Two OCCAM programs (running examples)

In the sequel, we only deal with one-sided simulations in which P' is constructed from P by replacing its fragments of code. In all proofs, we give a construction of P' which preserves parallelism of P (as the simulations are one-sided, we only insist on an injective mapping of parallel processes of P to P' ; e.g. additional parallel processes can be added to P' in a simulation. A programming language L is *at least as expressive* as a programming language L' , if there is an algorithm (compiler) which translates an arbitrary program P' in L' to a program P in L which simulates P' . Two programming languages L and L' are *equally expressive*, if L is at least as expressive as L' , and vice versa.

III. OCCAM WITH A SINGLE TOP-LEVEL `PAR` CONSTRUCTOR (OCCAM-1PAR)

Theorem 1: OCCAM with a single top-level `PAR` constructor (OCCAM-1PAR) is as expressive as OCCAM.

Proof: We need to show that an arbitrary OCCAM program can be simulated by an OCCAM program which uses exactly one `PAR` which is the top-level process. Consider an arbitrary OCCAM program. All channel declarations are moved to the top-level `PAR` (collisions of channel names are resolved by using fresh names). We will refer to processes of `PAR`s in the OCCAM program as child processes. Each child process is moved to the top-level `PAR`, together with declarations of all variables in its scope (including shared variables used in replicators). A new local integer variable `terminate` is declared in the child process; and two new channels are declared in the top-level `PAR`, we will call

them $ch.s$ (start) and $ch.e$ (end). A child process is started when it receives a message beginning with `FALSE` from the channel $ch.s$. This message also contains values of all the variables shared for reading between the parent and the child. When the child finishes its original program, it sends an acknowledgement to the channel $ch.e$. The parent (the sequence which replaces the `PAR`) starts its children and waits for their acknowledgements. Just before its own termination, the parent terminates its children. ■

Fig. 2 shows the translation of the program P_1 (Fig. 1) to an OCCAM-1PAR program P_3 with a single top-level `PAR`.

IV. OCCAM WITH SHARED CHANNELS (OCCAM-SH)

Recall that a channel in OCCAM connects two parallel processes. OCCAM-SH is an interesting variant in which channels are shared, i.e. a channel can be simultaneously accessed by arbitrarily many parallel processes for both reading and writing. When several `!` processes simultaneously write to a channel, then they are blocked until a `?` process reads from the channel. When several `?` processes simultaneously read from the same channel, they are blocked until a `!` process writes to the channel. When one or more `?` processes read from a channel and one or more `!` processes write to the channel, then eventually one of the `?` processes and one of the `!` processes are chosen for communication. This choice is made arbitrarily (i.e. the scheduler can freely decide which processes it chooses for communication). Then the message is passed from the chosen `!` process to the chosen `?` process and then these two processes terminate. Unlike in OCCAM, there is no `ALT` constructor in OCCAM-SH. Furthermore, OCCAM-SH programs use only one top-level `PAR` constructor.

It turns out that in spite of the absent `ALT` constructor, OCCAM-SH is a generalisation of OCCAM. We will show how an arbitrary OCCAM-1PAR program can be translated to an OCCAM-SH program which simulates the former one. This translation requires that the parallel processes and channels have identifiers. The order in which a parallel process appears in the single `PAR` constructor) will serve as its identifier. Analogously, channels are numbered in the order they are declared (to keep the notation simple, a channel identifier will serve as the channel's number). Let the numbering start with 0, let N denote the number of processes.

Theorem 2: OCCAM-SH is at least as expressive as OCCAM.

Proof: We have already proved that an arbitrary OCCAM program can be simulated by an OCCAM-1PAR program. It remains to show how the `ALT` constructors of OCCAM-1PAR are simulated in OCCAM-SH. Consider an OCCAM-1PAR program. For each parallel process p , merge all the channels from which the process reads to a single channel $ch.in_p$. This shared channel will be the only one which the process p will read, and p will be its only reader (there may be more than one writer, though). Declare arrays `INT pending.ch[N]` and `MSG pending.msg[N]` in the scope of each parallel process, where `MSG` is the type of messages transferred in the OCCAM-1PAR program. Initialise

Program P_3

```

1: CHAN ch1, ch2:
2: CHAN ch.s1, ch.e1, ch.s2, ch.e2, ch.s3, ch.e3:
3: PAR
4:   SEQ -- original top-level code
5:   SEQ i = 0 FOR 2
6:     BOOL ack:
7:     SEQ -- replacement of parent PAR
8:       ch.s1 ! FALSE; i -- start child 1
9:       ch.s2 ! FALSE; i -- start child 2
10:      ch.s3 ! FALSE; i -- start child 3
11:      ch.e1 ? ack -- wait for end of child 1
12:      ch.e2 ? ack -- wait for end of child 2
13:      ch.e3 ? ack -- wait for end of child 3
14:      ch.s1 ! TRUE; 0 -- terminate child 1
15:      ch.s2 ! TRUE; 0 -- terminate child 2
16:      ch.s3 ! TRUE; 0 -- terminate child 3
17:    BOOL terminate:
18:    INT i:
19:    SEQ -- child 1 of PAR
20:      ch.s1 ? terminate; i -- wait for parent
21:      WHILE NOT terminate
22:        SEQ
23:          INT m:
24:          SEQ j = 0 FOR 2
25:            ALT
26:              ch1 ? m
27:              SKIP
28:              ch2 ? m
29:              SKIP
30:            ch.e1 ! TRUE -- acknowledge parent
31:            ch.s1 ? terminate; i -- wait for parent
32:          BOOL terminate:
33:          INT i:
34:          SEQ -- child 2 of PAR
35:            ch.s2 ? terminate; i -- wait for parent
36:            WHILE NOT terminate
37:              SEQ
38:                ch1 ! 2 * i
39:                ch.e2 ! TRUE -- acknowledge parent
40:                ch.s2 ? terminate; i -- wait for parent
41:          BOOL terminate:
42:          INT i:
43:          SEQ -- child 3 of PAR
44:            ch.s3 ? terminate; i -- wait for parent
45:            WHILE NOT terminate
46:              SEQ
47:                ch2 ! (2 * i) + 1
48:                ch.e3 ! TRUE -- acknowledge parent
49:                ch.s3 ? terminate; i -- wait for parent

```

Fig. 2. OCCAM \rightarrow OCCAM-1PAR ($P_1 \rightarrow P_3$)

the array `pending.ch[N]` with values `-1`, i.e. insert the following sequence after the initial `SEQ` in each process:

```

SEQ i = 1 FOR NP
  pending.ch[i] := -1

```

Replace each `ch ! m` of the parallel process w with

```

BOOL ack:
SEQ
  ch.in_r ! w; ch; m
  ch.in_w ? ack

```

where r is the identifier of the process which reads the channel ch in the original OCCAM program.

Replace each

ALT

$ch_0 ? m$
 b_0

...

$ch_k ? m$
 b_k

of the parallel process r with the sequence

INT w, ch :

BOOL consumable:

SEQ

consumable := FALSE

WHILE NOT consumable

SEQ

$w := 0$ -- look for a consumable message

WHILE ($w < NP$) AND ($pending.ch[w] \neq ch_0$) ...

AND ($pending.ch[w] \neq ch_k$)

$w := w + 1$

IF

$w < NP$

SEQ -- found a consumable message

$m := pending.msg[w]$

$ch := pending.ch[w]$

consumable := TRUE

TRUE -- otherwise save another message

SEQ

$ch.in_r ? w; ch; m$

$pending.ch[w] := ch$

$pending.msg[w] := m$

IF -- execute the corresponding branch of **ALT**

$ch = ch_0$

SEQ

$ch.in_w ! TRUE$

$pending.ch[w] := -1$

b_0

...

$ch = ch_k$

SEQ

$ch.in_w ! TRUE$

$pending.ch[w] := -1$

b_k

Treat each $ch ? m$ as

ALT

$ch ? m$
SKIP

and use the translation above.

Hence, each $!$ subsequently blocks until it is acknowledged by the **ALT**ing process. The **ALT**ing process reads all incoming messages, but acknowledges only those which have been consumed by the **ALT** of the **OCCAM-IPAR** program. ■

Theorem 3: **OCCAM** is at least as expressive as **OCCAM-SH**.

Proof: We present a construction which replaces shared channels with directed channels which connect exactly two processes. Let N denote the number of parallel processes in the **OCCAM-SH** program. For each shared channel ch , an additional parallel process $sh.ch$ is created which relays the communication on the shared channel using **OCCAM** channels. These $sh.ch$ processes terminate when all the other parallel processes terminate. A process $sh.ch$ is connected via three channels with each parallel process p ($p = 0, \dots, N-1$) of the **OCCAM-SH** program: $ch.r_{sh.ch}[p]$, $ch.w_{sh.ch}[p]$ and

$ch.d_{sh.ch}[p]$. All these channels are declared as global. The first two channels are oriented towards $sh.ch$, the third one towards the parallel process p .

The program of a process $sh.ch$ is:

BOOL terminate:

INT out, t.count:

MSG m:

SEQ

t.count := 0

WHILE t.count < N

SEQ

ALT -- pick a reader

$ch.r_{sh.ch}[0] ? terminate$

out := 0

...

$ch.r_{sh.ch}[N-1] ? terminate$

out := N - 1

IF

terminate

t.count := t.count + 1

TRUE -- otherwise

ALT -- pick a writer, deliver m to the reader

$ch.w_{sh.ch}[0] ? m$

$ch.d_{sh.ch}[out] ! m$

...

$ch.w_{sh.ch}[N-1] ? m$

$ch.d_{sh.ch}[out] ! m$

Replace each $ch ? m$ of the parallel process r in the **OCCAM-SH** program with the sequence

SEQ

$ch.r_{sh.ch}[r] ! FALSE$

$ch.d_{sh.ch}[r] ? m$

Replace each $ch ! m$ of the parallel process w with $ch.w_{sh.ch}[w] ! m$.

Insert $ch.r_{sh.ch}[p] ! TRUE$ at the end of each parallel process p of the **OCCAM** program. ■

V. CHANNEL-LESS OCCAM (OCCAM-CL)

OCCAM-SH1 is a stricter variant of **OCCAM-SH** in which each parallel process p is allowed to read only from one channel $ch.in_p$, whereby p is the only process which reads from the channel $ch.in_p$. We call this variant channel-less, because the identifiers of channels unify with the identifiers of processes. The processes can be numbered in the order they appear in the single **PAR** constructor.

Theorem 4: **OCCAM-SH1** and **OCCAM-SH** are equally expressive.

Proof: An **OCCAM-SH1** program is also an **OCCAM-SH** program. Conversely, consider an arbitrary **OCCAM-SH** program. Translate it to **OCCAM** and then back to **OCCAM-SH** using compilers from the proofs of Theorem 3 and Theorem 2. This yields an **OCCAM-SH1** program. ■

OCCAM-CL syntactically removes the redundancy related to channels from **OCCAM-SH1**. An arbitrary **OCCAM-SH1** program can be rewritten to **OCCAM-CL** as follows:

- Remove all channel declarations.
- Replace each $ch ? m$ with $? m$.
- Replace each $ch ! m$ with $p ! m$, where p is the identifier of the process which reads the channel ch .

Figure 3 illustrates the correspondence between OCCAM-SH1 and OCCAM-CL.

<pre> 1: Program P₄ 2: CHAN ch0, ch1, ch2: 3: PAR 4: INT m: 5: SEQ -- process 0 6: SEQ i = 0 FOR 2 7: SEQ j = 0 FOR 2 8: ch0 ? m 9: ch1 ! TRUE 10: ch2 ! TRUE 11: BOOL ack: 12: SEQ -- process 1 13: SEQ i = 0 FOR 2 14: SEQ 15: ch0 ! 2 * i 16: ch1 ? ack 17: BOOL ack: 18: SEQ -- process 2 19: SEQ i = 0 FOR 2 20: SEQ 21: ch0 ! (2 * i) + 1 22: ch2 ? ack </pre>	<pre> 1: Program P₅ 2: 3: PAR 4: INT m: 5: SEQ -- process 0 6: SEQ i = 0 FOR 2 7: SEQ j = 0 FOR 2 8: ? m 9: 1 ! TRUE 10: 2 ! TRUE 11: BOOL ack: 12: SEQ -- process 1 13: SEQ i = 0 FOR 2 14: SEQ 15: 0 ! 2 * i 16: ? ack 17: BOOL ack: 18: SEQ -- process 2 19: SEQ i = 0 FOR 2 20: SEQ 21: 0 ! (2 * i) + 1 22: ? ack </pre>
--	--

Fig. 3. Hand-made translations of the OCCAM program P_1 (Fig 1) to OCCAM-SH1 (left) and OCCAM-CL (right)

VI. CONCLUSIONS

We proposed a programming language OCCAM-CL which differs from OCCAM (only) in having no nested PAR processes, no ALT constructors, and—most importantly—no channels. In spite of this, OCCAM-CL is as expressive as OCCAM. We proved this using several OCCAM variants and compilers which translate programs from one variant to any other one. These compilers preserve parallelism of programs as well as their message complexity (up to a constant multiplicative factor). A similar result was published in [25] for a lambda calculus with typed asynchronous channels and a lambda calculus with typed actors.

When it comes to writing an actual compiler, the choice between OCCAM-CL and OCCAM is not just a matter of taste. Apparently, writing a parser for OCCAM-CL is easier, but there are more subtle reasons for favouring OCCAM-CL. For example, OCCAM requires that each channel connects exactly two parallel processes (one reader, one writer). However, its syntax does not prevent the programmer from violating this requirement. It is up to the compiler or a run-time system to detect such a violation.

The channel-less approach can be found in actor models [26] as well as in contemporary programming languages, e.g. Erlang and Akka. A subsequent extension of Akka with the channel concept is in the light of our results a backward step. It does not increase expressiveness, unnecessarily increases the complexity of the language and the compiler, and increases the structural complexity of programs which mix the channel and channel-less paradigms.

REFERENCES

- [1] C. A. R. Hoare, *Communicating Sequential Processes*. Prentice Hall, 1985.
- [2] R. Milner, “Elements of interaction,” *Communications of the ACM*, vol. 36, no. 1, pp. 70–89, 1993, Turing Award lecture.
- [3] M. Ahuja, A. D. Kshemkalyani, and T. Carlson, “A basic unit of computation in distributed systems,” in *International Conference on Distributed Computing Systems (ICDCS)*. IEEE Computer Society, 1990. doi: 10.1109/ICDCS.1990.89327 pp. 12–19.
- [4] J. Biernacki, “Alvis models of safety critical systems state-base verification with nuXmv,” in *FedCSIS*, ser. Annals of Computer Science and Information Systems, vol. 8. IEEE, 2016. doi: 0.15439/2016F264 pp. 1701–1708.
- [5] SGS Thomson Ltd., *OCCAM 2.1 Reference Manual*. Prentice Hall, 1988.
- [6] A. Ripke, A. A. Allen, R. Alastair, and Y. Feng, “Distributed computing using channel communications in Java,” in *Communicating Process Architectures 2000*. IOS Press, 2000, pp. 49–62.
- [7] P. H. Welch and F. R. M. Barnes, “Communicating mobile processes: Introducing OCCAM- π ,” in *Communicating Sequential Processes*, ser. LNCS. Springer, 2005, vol. 3525, pp. 712–713.
- [8] N. C. C. Brown, “Rain: A new concurrent process-oriented programming language,” in *Communicating Process Architectures*. IOS Press, 2006, pp. 237–251.
- [9] R. Loogen, “Eden—parallel functional programming with Haskell,” in *Central European Functional Programming School, (CEFP), Budapest, Hungary*, ser. LNCS, vol. 7241. Springer, 2011. doi: 10.1007/978-3-642-32096-5_4 pp. 142–206.
- [10] G. D’Angelo, S. Ferretti, and M. Marzolla, “Time warp on the Go,” in *Proc. of the International ICST Conference on Simulation Tools and Techniques*, ser. SIMUTOOLS ’12. ICST, Brussels, Belgium, 2012. doi: 10.5555/2263019.2263057 pp. 242–248.
- [11] M. W. Heath, W. P. Bursleson, and I. G. Harris, “Synchro-tokens: A deterministic GALS methodology for chip-level debug and test,” *IEEE Transactions on Computers*, vol. 54, no. 12, pp. 1532–1546, 2005. doi: http://doi.ieeecomputersociety.org/10.1109/TC.2005.203
- [12] M. A. Rahimian, S. Mohammadi, and M. Fattah, “A high-throughput, metastability-free GALS channel based on pausable clock method,” in *Asia Symposium on Quality Electronic Design*. IEEE, 2010. doi: 10.1109/ASQED.2010.5548259 pp. 294–300.
- [13] P. Hajder, L. Rauch, M. Nycz, and M. Hajder, “A heterogeneous parallel processing system based on virtual multi-bus connection network,” in *FedCSIS (Position Papers)*, ser. Annals of Computer Science and Information Systems, vol. 19, 2019. doi: 10.15439/2019F356 pp. 9–17.
- [14] *ISO/IEC 9945-1: 1990 Information Technology. Portable Operating System Interface (POSIX), Part 1: System Application Program Interface*.
- [15] J. Galletly, *OCCAM 2. Including OCCAM 2.1*. UCL Press, 1996.
- [16] A. W. Roscoe, “Denotational semantics for OCCAM,” in *Seminar on Concurrency, Carnegie-Mellon University*. London, UK: Springer, 1985, pp. 306–329.
- [17] A. Eliëns, “Semantics for OCCAM,” Centre for Mathematics and Computer Science (CWI), Amsterdam, Tech. Rep. 6255, 1986.
- [18] Y. Gurevich and L. S. Moss, “Algebraic operational semantics and OCCAM,” in *Proceedings of the 3rd Workshop on Computer Science Logic, ser. CSL ’89*. London, UK: Springer, 1990. doi: 10.1007/3-540-52753-2_39 pp. 176–192.
- [19] A. W. Roscoe, M. H. Goldsmith, and B. G. O. Scott, “Denotational semantics for OCCAM 2, part 1,” *Transputer Communications*, vol. 1, pp. 65–91, 1994.
- [20] —, “Denotational semantics for OCCAM 2, part 2,” *Transputer Communications*, vol. 2, pp. 25–67, 1994.
- [21] MPI Forum, *MPI-4.0*. HLRS, 2021.
- [22] F. Cesarini and S. Thompson, *ERLANG Programming*. O’Reilly, 2009.
- [23] D. Wyatt, *Akka Concurrency*. Artima Incorporation, 2013.
- [24] I. Graham and T. King, *The Transputer Handbook*. Prentice Hall, 1990.
- [25] S. Fowler, S. Lindley, and P. Wadler, “Mixing metaphors: Actors as channels and channels as actors,” in *31st European Conference on Object-Oriented Programming, ECOOP 2017, Barcelona, Spain*, ser. LIPICs, vol. 74. Schloss Dagstuhl—Leibniz-Zentrum für Informatik, 2017. doi: 10.4230/LIPICs.ECOOP.2017.11 pp. 11:1–11:28.
- [26] C. Hewitt, P. Bishop, and R. Steiger, “A universal modular actor formalism for artificial intelligence,” in *Proceedings of the 3rd International Joint Conference on Artificial Intelligence*, ser. IJCAI’73. Morgan Kaufmann, 1973, pp. 235–245.

Improvement of design anti-pattern detection with spatio-temporal rules in the software development process

Łukasz Puławski

Faculty of Mathematics, Informatics and Mechanics, University of Warsaw

Email: lpulawski@mail.mimuw.edu.pl

Abstract—In [1] we presented a framework for mining spatio-temporal rules in the software development process. The rules are based on specific relations between structures of the source code which relate both to spatial (e.g. a direct call between methods of two classes) and temporal dependencies (e.g. one class introduced into the source code before the other) observed in the process. To some extent, spatio-temporal rules allow us to predict where and when certain design anti-patterns will appear in the source code of a software system. This paper presents how, with slight modifications, such framework can be used to improve the quality of detecting a few popular design anti-patterns, such as Blob, Swiss Army Knife, YoYo or Brain Class. In the proposed method, we not only check the structure of a piece of the source code, but we also analyse its spatio-temporal relations. Only on the basis of the two analyses can we decide if the given piece of code is an anti-pattern. Experimental validation shows that the addition of spatio-temporal perspective improves detection of anti-patterns by 4% in terms of F-measure.

I. INTRODUCTION

DESIGN *anti-pattern* is a commonly used, bad solution for a recurring problem in software design. Software developers, when faced with a common design problem, tend to reinvent the solutions that are well-known for their bad properties and widely discussed in available literature (see [2]). This phenomenon is definitely due to many sophisticated reasons, which are discussed in a variety of scientific and popular publications (see [3]). The following paragraphs give a few examples of design-anti-patterns.

Base bean is an anti-pattern in object-oriented design, where the base class is a collection of numerous utility methods used by its subclasses. Such a design breaks some fundamental concepts of object-oriented programming: The relation between subclass and superclass does not resemble the actual domain model, the superclass has many responsibilities and it usually does not store any state useful for subclasses.

Brain class and God Class are similar anti-patterns that refer to classes that provide too much complexity and tend to centralize logic of some area of the system. The difference between God class and Brain class is that the former is a large controller class that depends on data from the surrounding classes, whereas the latter does not use the data from other classes, tends to be more cohesive and encapsulates the logic in its own complex methods (see [4], [5], [6], [7]).

Swiss Army Knife, (abbreviated as SAK) is an excessively complex class with numerous unrelated utility methods. It

tends to appear when the creator attempts to provide a routine for all possible uses of the class or make a single class serve many complex unrelated functions (see [8], [9] and [6]).

YoYo is an anti-pattern in which the flow control is scattered over complicated inheritance structure, so that in order to understand the algorithm in the source code, one has to switch between many classes within a common inheritance tree (see [10], [11], [6]).

Design anti-patterns make the software more complex, harder to maintain and defect-prone (see [12], [13], [14], [15]). This is why their detection is a primary concern in software engineering.

II. DESIGN ANTI-PATTERN DETECTION

The objective of design anti-patterns detection is to provide an automated method for the discovery of fragments of the program source code which constitute a design anti-pattern. To make this task more formal we will use graph-theoretical terms.

A. Software as a graph

We can treat the source code of the system written in Java as a multigraph called *software snapshot* according to the following rules: The nodes of the multigraph are all the *source code entities*, namely: packages, interfaces, classes, fields, methods (including constructors). They can be connected by labeled edges according to the following rules:

- There is an edge (e_1, e_2) labeled 'contain' iff the source code of the entity represented by e_2 is contained in the source code of the entity represented by e_1 (we will assume that classes and interfaces are contained in packages),
- There is an edge (e, c) labeled 'variable' iff the body of the entity represented by e declares at least one variable of the type represented by class c . Each such declaration corresponds to a single edge.
- There is an edge (m, c) labeled 'parameter' iff the method represented by entity m declares at least one formal parameter of the type represented by c . Each such declaration corresponds to a single edge.
- There is an edge (c_1, c_2) labeled 'extend' iff the class represented by c_1 is a direct subclass of the class represented by c_2 or the class represented by c_1 is an implementation

of the interface represented by c_2 . Each such class extension or interface implementation corresponds to a single edge.

- There is an edge (e, m) labeled 'call' iff the body of the entity represented by e contains at least one call of a method represented by the method entity m . Each such call corresponds to a single edge.
- There is an edge (e, f) labeled 'refer' iff the body of the entity represented by e contains at least one reference to the field represented by the field entity f . Each such reference corresponds to a single edge.
- There is an edge (f, c) labeled 'type' iff c represents a class that is a declared type of the field represented by f or a declared return type of a method declared by f .

Additionally, each node of the graph may be described by a set of applicable *software metrics* that measure its complexity (see [16]). For greater consistency we will assume that the values of metrics form a vector of additional labels of the node. Consequently, we can treat a software snapshot as a node-labeled and edge-labeled multigraph. This allows us to remain in the graph-theoretical domain. The list of metrics that are used is given in the following subsection:

B. Software metrics

- **Data abstraction coupling**
This metric is applicable for class entities and measures how many instances of other classes are instantiated within the source code of a given class.
- **Fan out**
This metric is similar to *Data abstraction coupling*, which measures the number of classes a given class depends on.
- **Cyclomatic complexity and NPath complexity**
These two metrics measure the complexity of a code block. Cyclomatic complexity, based on the classic work [17], denotes the number of decision point instructions within the body block increased by 1. The NPath complexity (see [18]) denotes the theoretical maximum number of different acyclic execution paths that could go through the code block.
- **NCSS - the number of lines of the source code in each entity (file, class, method)**
This simple metric measures how many lines of code a given file, class or method take. Technically, the evaluations do not count empty lines or lines with comments, so that it approximates the actual size of the respective source code fragment.
- **Lack of cohesion of methods (LCOM1, LCOM2, LCOM3, LCOM4, TCC)**
LCOM is a suite of software metrics that evaluate the design of a given class by quantitative analysis of relations between its methods and attributes (see [19]). LCOM1 is defined as the difference between the powers of two sets: the set of all pairs of different methods that use a non-empty disjoint set of class attributes and the set of all pairs of different methods that use at least one

attribute altogether. If the result is negative, the value of this metric is set to 0.

LCOM2 and LCOM3 metrics are defined for the class and the formulae to evaluate them, are based on the following notions: m - the number of methods in a class, A - the set of attributes of a class, m_a - the number of methods that access attribute a :

$$LCOM2 = 1 - \frac{\sum_{a \in A} m_a}{m * |A|}$$

$$LCOM3 = \frac{m - \frac{1}{|A|} \sum_{a \in A} m_a}{m - 1}$$

LCOM2 corresponds to the fraction of methods that do not access a specific attribute normalized over all attributes. LCOM3 is similarly normalized with respect to attributes and methods and its value may range from 0 to 2.

LCOM4 (see [20]) is expressed in terms of a graph of inter-method dependencies. We say that two methods are dependent iff one of them calls the other one or there is at least one attribute used by both methods. The number of connected components in such a graph is the value of LCOM4. Instead of looking at relations between methods and attributes within a single class, we can in a similar manner measure the relation between methods of a given class, with the use of Tight Class Cohesion (TCC) metric (see [21]). TCC is defined as the number of pairs of methods that invoke one another divided by the number of all such pairs.

- **Depth of inheritance tree (DIT)**
This metric is applicable to classes only. It measures the number of nodes on the path in the inheritance tree from the node representing the `java.lang.Object` class to the node representing the given class. Large value of this metric indicates a deep inheritance tree, which might indicate the presence of a *Yo-yo* design anti-pattern.
- **Fan in (FI)**
A metric dual to the *Fan out*. It measures the number of other classes that depend on a given class. The greater the value, the more likely it is that a change in the class will affect other fragments of the software source code.

In this context we can formalize the problem of detecting design anti-patterns as the problem of finding all such subgraphs of the software snapshot that correspond to instances of these anti-patterns. For practical reasons we will only consider subgraphs that satisfy the following Definition 1:

Definition 1 (containment-completeness): Let $SSn = (V, E)$ be a software snapshot, where V is a set of nodes and E is a multi-set of labeled edges. We will say that subgraph $g = (V_g, E_g)$ of SSn is *containment - complete*, if for any $n_1 \in V_g$, if there is a node $n_2 \in V$ such that n_2 is connected by 'contain' edge with n_1 in SSn then $n_2 \in V_g$.

We apply this constraint on the subgraphs, since we want the analysed subgraphs to resemble a consistent fragment of the source code with its natural hierarchical structure. For

example, if we take a subgraph with a node that corresponds to a class, we also want methods and fields of this class to be part of the subgraph. Therefore, in all the following considerations a *subgraph* always means a *containment-complete subgraph*.

C. Detectors of design anti-patterns

The following paragraphs provide a semi-formal description of method of detecting specific design anti-patterns used in this research. Each detection strategy is derived from existing research mentioned in the respective subsections below, but they are adapted to the graph-theoretical model described in Section II-A. In order to enhance comprehension, detection methods are described in mixed graph-theory and software-engineering terms. Each description can be translated to purely graph-theoretical terms so that for any subgraph of a software snapshot we can always tell if it satisfies the conditions described below. A exemplary rationale on how the conceptual definition can be translated to graph-theoretic model is given for the first detector only. Similar reasoning for other types of anti-patterns can be found in the articles referred to in the respective following subsections.

1) *Swiss army knife*: A *Swiss Army Knife (SAK)* for short, is an excessively complex class interface. It is present when e.g. the creator attempts to provide a method for all possible uses of the class or make a single class serve many complex unrelated functions. Methods described or referenced in [8], [9], [6] and [22] provide a semi-formal description of the pattern as *a class with many unrelated methods with high complexity which implements many interfaces*. Clearly, this can be translated to the graph-theoretical language in the context of a software snapshot: a class with many interfaces corresponds to each node such that paths which start from it and contain edges of type 'extend' and 'implement' reach many nodes which represent interface entity. A complex method is simply a method with high values of the complexity metrics such as *NPath complexity* or *cyclomatic complexity*. Additionally, if a class is intended to serve many purposes, one can expect that it has methods that are being called by many other classes. This conceptual description can be translated in the formal graph-theoretical definition described below:

Definition 2 (foreign call): Let c_1 and c_2 be two classes such that they are not connected by a path build from edges labeled 'extend'. Every call from a method contained in c_2 is a *foreign call* for class c_1 .

Swiss Army Knife is each class c that satisfies the following conditions:

- c has more than 6 methods,
- the value of metric LOC for c exceeds 150,
- the sum of cyclomatic complexity of methods contained in c exceeds 30,
- the sum of NPath complexity of methods contained in c exceeds 120,
- the number of non-trivial methods contained in c multiplied by the average NPath complexity of such methods exceeds 160,

- the number of methods called by a foreign call exceeds 2,
- the number of foreign calls exceeds 7.

2) *Anemic entities*: Anemic entities are classes which only store data and do not provide any functionality (see [23], [24]). A straightforward, naive approach for detecting this pattern is to take classes which have:

- many fields,
- only accessor methods (i.e. methods with a single line of code, which refer to only a single field and have cyclomatic and NPath complexity equal to one),
- default and possibly an initializing constructor.

This heuristic turns out to be inaccurate. Therefore, we need to define two notions: an *effectively trivial method* and a *complex constructor*.

A method m is *effectively trivial* if :

- The class c in which m is contained has field f of type t such that m refers to f and either m has 1 argument of type t and *void* return type or it has 0 arguments and return type t ,
- m has at most 5 lines of code,
- m has cyclomatic complexity not greater than 3.

The constructor con contained in c is complex iff:

- The number of arguments of con exceeds the number of fields contained in c ,
- the lines of code in con exceeds 150% of the number of fields contained in c ,
- the cyclomatic complexity of con exceeds 150% of the number of fields contained in c .

This allows us to provide a formal definition for anemic entity: A class is an *Anemic Entity* iff:

- it has more than 8 fields,
- it has more than 8 methods,
- all methods but one are trivial or *effectively trivial*,
- there are no complex constructors contained in c ,
- all subclasses of c satisfy the above four conditions.

3) *Blob (also known as God Class)*: (see [4], [5], [6], [7]). Entity c is considered a Blob iff:

- c calls more than 7 effectively trivial methods of other classes,
- proportion of the number of effectively trivial methods of other classes called by c to the number of other methods of other classes called by c exceeds 0.6,
- the sum of cyclomatic complexity of its non-trivial methods exceeds 55,
- the value of the metric TCC does not exceed 0.3.

4) *Brain class*: (see [4], [5], [6], [7]). Entity c is considered brain class iff

- c is not a God Class, as defined in the preceding subsection,
- c has more than 2 non-trivial methods with more than 4 outgoing edges of type 'call' and more than 15 lines of code (controller methods),
- c calls more than 5 trivial methods of other classes,

- proportion of the number of trivial methods of other classes called by c to the number of non-trivial methods of other classes called by c does not exceed 0.6,
- the number of calls to trivial methods divided by the number of lines of the source of c is smaller than 0.2,
- the value of tight class cohesion metric for c does not exceed 0.5,
- One of the following conditions is true:
 - sum of cyclomatic complexity of methods contained in c exceeds 50 and the value of NCSS metric for c exceeds 400,
 - sum of cyclomatic complexity of methods contained in c exceeds 90 and the value of NCSS metric for c exceeds 50,

5) *Base Bean*: (see [25]) *Base Bean* is a class which only provides utility methods for its subclasses.

A method m contained in class c is an *utility method* iff:

- m is neither a constructor nor a trivial method,
- m does not refer to any field contained in c , nor to a field in any direct or indirect superclass of c ,
- there is no path that connects m with field f contained in c or one of its direct or indirect superclasses, such that the last edge on this path has a label 'refer' and all other edges have a label 'call'.

Conceptually, a utility method is a non-trivial method that does not modify the state nor does it orchestrate other methods contained in the class it is defined in or any of its ancestors and is used only by its descendants.

A class c is *Base Bean* iff:

- it has more than 2 utility methods,
- c has at least 5 direct or indirect subclasses,
- the number of incoming edges of type 'call' from the hierarchy of c to utility methods contained in c exceeds 2.

6) *Yo-yo*: (see [10], [11], [6]). A containment-complete sub-graph induced by nodes $Y = \{e_1, \dots, e_n\}$ is a *YoYo* iff:

- Each pair $(e_i, e_j) \in Y \times Y$ is connected by a path constructed from edges of type 'extend', where each edge is treated as undirected,
- the longest path between any two nodes from Y constructed from such edges exceeds 5,
- the number of edges (m_1, m_2) with label 'call' or 'refer' such that m_2 is not a trivial method and there are edges $(m_1, e_i), (m_2, e_j)$ with label 'contain', $i \neq j$ exceeds 5.
- there is no super-set of nodes $Y' \supseteq Y$ such that graph induced by Y' satisfies the above three conditions.

7) *Data Clumps*: (see [15], [26], [27], [28]) a *Data Clump* is an anti-pattern that occurs when a group of data items are being passed together in the source code. This informal definition can be rephrased formally:

Let $parameters(m)$ denote set of entities connected with m by an edge labeled 'parameter'. A set of method entities $M = \{m_1, \dots, m_n\}$ is a *Data Clump* iff:

- n exceeds 3,

- $|parameters(m_i)|$ exceeds 3 for each i ,
- for each pair (m_i, m_j) such that $i \neq j$ and m_i and m_j are connected by 'call' edge, $|parameters(m_i) \cap parameters(m_j)| = \min(|parameters(m_i)|, |parameters(m_j)|)$,
- there is no such superset of M that satisfies the above conditions.

8) *Circular dependency*: *Circular dependency* is a relation between two or more software entities transitively contained in different packages which either call each other directly or indirectly to function properly. We can translate this into graph theoretical terms:

A pair of classes (c_1, c_2) forms a circular dependency iff:

- there exist two different packages p_1, p_2 such that there are edges $(c_1, p_1) (c_2, p_2)$ with label 'contain',
- there are two methods m_1, m_2 , such that there are edges (m_1, c_1) and (m_2, c_2) with label 'contain',
- there is a path build only from edges labeled 'call' from m_1 to m_2 and another such path from m_2 to m_1 .

9) *Detection quality*: The detection quality of purely static methods of identification of design anti-patterns described in the preceding subsections is presented in Table I.

	SAK	BI	DC	BB
Argo Uml	0.78/1.0	0.90/0.76	N/A	0.71/0.88
Elasticsearch	0.78/0.99	0.83/0.19	0.99/0.96	0.71/0.88
JHotDraw	1.0/0.91	1.0/1.0	0.28/0.0	N/A
Lucene	0.86/1.0	0.88/0.9	N/A	0.97/1.0
Struts	0.99/1.0	N/A	0.98/0.1	N/A
Wildfly	0.94/1.0	0.92/1.0	0.99/1.0	1.0/1.0
Xerces	0.8/0.89	0.91/0.69	0.99/0.84	N/A

	BC	YY	AE
Argo Uml	N/A	1.0/1.0	1.0/1.0
Elasticsearch	0.87/0.84	0.98/1.0	1.0/1.0
JHotDraw	0.0/0.0	N/A	N/A
Lucene	0.95/0.78	1.0/1.0	N/A
Struts	N/A	N/A	N/A
Wildfly	N/A	N/A	1.0/1.0
Xerces	N/A	0.98/1.0	N/A

Table I

THE TABLE SHOWS THE QUALITY OF DETECTION OF INSTANCES OF THE DESIGN ANTI-PATTERNS. THE COLUMNS CORRESPOND TO A RANGE OF ANTI-PATTERN TYPES WHEREAS THE ROWS PRESENT DIFFERENT SOFTWARE SYSTEMS. THE DETECTION METHODS ARE DESCRIBED IN SUBSECTIONS II-C1–II-C8 ABOVE. EACH CELL CONTAINS TWO NUMBERS: PRECISION/RECALL. SAK = SWISS ARMY KNIFE, BI = BLOB, DC = DATA CLUMPS, BB = BASE BEAN, BC = BRAIN CLASS, YY = YoYo, AE = ANEMIC ENTITY.

III. SPATIAL RELATIONS

Since design anti-patterns are subgraphs of one common graph, we can introduce the notion of distance between two patterns defined as the length of the shortest path that connects nodes from these subgraphs:

Definition 3 (closeness and remoteness of patterns): Let $PI_1 = (V_1, E_1)$ and $PI_2 = (V_2, E_2)$ be subgraphs of software snapshot SSn . We will say that

PI_1 and PI_2 are d -distance-close iff $d(PI_1, PI_2) \leq d$, where $d(PI_1, PI_2, SSn) = \min_{v_1 \in V_1, v_2 \in V_2} dist(v_1, v_2, SSn)$ where $dist(a, b, G)$ is the distance between vertices a and b measured as the shortest path between them in the multigraph G treated as undirected graph.¹

¹This makes dist symmetric.

Similarly: PI_1 and PI_2 are *d-distance-remote* iff $d(PI_1, PI_2) > d$.

IV. SOFTWARE EVOLUTION

Usually software development is done in the *source code management* system that allows us to track all changes done to the source code. Each individual modification of the source code is called *commit* and is identified by unique number called *revision*. Commit has precise date, author and a set of modifications to the source code files. If we take the commits from a single main development branch, we can order them linearly according to the commit date. The code at each revision has a corresponding software snapshot, thus we can treat linearly-ordered sequence of such snapshots as a model of *software evolution*.

One specific design anti-pattern can be observed at multiple revisions. The set of all such revisions will be called the *lifespan* of a pattern. Clearly, the lifespan can be divided into intervals of maximum lengths such that the corresponding pattern instance is not observed in the revision that directly precedes the left end of this interval nor is it observed in the revision that directly follows the right end of this interval. Each such interval will be called *occurrence* of the pattern. Please note that each pattern instance can potentially have more than one occurrence, as e.g. a certain software structure can be removed and then added again to the source code.

V. SPATIO-TEMPORAL RELATIONS

If we take two different patterns PI_1 and PI_2 and their two occurrences $l_1 = (l_{start}^1, l_{end}^1)$ and $l_2 = (l_{start}^2, l_{end}^2)$, we can tell the *temporal* relation between l_1 and l_2 (e.g. l_1 may directly precede l_2 when $l_{end}^1 = l_{start}^2$). In order to model the temporal relations we use Allens interval algebra in this research (see [29]), which introduces 13 different possible relations which comprise equality and 6 pairs of invertible relations.

We will say that Allens relation between l_1 and l_2 defined above is *non-inverted* iff $l_{start}^1 < l_{start}^2 \vee (l_{start}^1 = l_{start}^2 \wedge l_{end}^1 < l_{end}^2)$. Conceptually it means that l_1 *takes place before* some non-degenerated sub-interval of l_2 . In other words, l_2 will last for some time after l_1 has started. The non-inverted relations of these pairs are given in the following list:

- 1) l_1 *takes place before* l_2 if there exists a revision s such that $(l_{end}^1 < s < l_{start}^2)$
- 2) l_1 *meets* l_2 ($l_{end}^1 = l_{start}^2$)
- 3) l_1 *overlaps* l_2 ($l_{start}^1 < l_{start}^2 < l_{end}^1 < l_{end}^2$)
- 4) l_1 *starts* l_2 ($l_{start}^1 = l_{start}^2 \wedge l_{end}^1 < l_{end}^2$)
- 5) l_1 *contains* l_2 ($l_{start}^1 < l_{start}^2 < l_{end}^1 < l_{end}^2$)
- 6) l_1 *is finished by* l_2 ($l_{start}^1 < l_{start}^2 \wedge l_{end}^1 = l_{end}^2$)

Each Allens operator A has its inversion A^{-1} (which is also an Allens operator) defined by: xAy iff $yA^{-1}x$.

Let $0 < d_c < d_r$ be two fixed natural numbers which we will associate with d_c -distance-closeness and d_r -distance-remoteness relation respectively. If l_1 and l_2 are in A Allen relation and at some revision graph induced by nodes of PI_1 is d_c -distance-close to graph induced by nodes of PI_2 , then

we will say that these two occurrences are in A - d_c -distance-closeness *spatio-temporal relation*. If these graphs are d_r -distance-remote at all revisions we will say that they are in A - d_r -distance-remoteness *spatio-temporal relation*. Please note that the above definitions are also valid if PI_1 and PI_2 are never observed together at a single revision.

If we take a single occurrence l_1 of some anti-pattern PI_1 ($[l_1, PI_1]$), we can tell all its spatio-temporal relations to all other occurrences of other anti-patterns. Each such relation can be characterized by three arguments:

- T - the type of other anti-pattern ($T \in \{\text{BLOB, SAK, Base Bean, YOYO, Brain Class, Data Clump, Anemic Entity, Circular Dependency}\}$),
- A - the non-inverted Allen algebra relation ($A \in \{\text{takes place before, meets, overlaps, starts, contains, is finished by}\}$) and
- $s \in \{\text{remote, close}\}$ - which determines if we are talking about A - d_c -distance-closeness or A - d_r -distance-remoteness *spatio-temporal relation*.

Therefore, for each triplet (T, A, s) we can tell how many respective spatio-temporal relations to $[l_1, PI_1]$ exist in the software evolution. This yields a vector in $\mathcal{N}^{(8 \times 6 \times 2)}$ space which provides information about the number of all spatio-temporal relations of occurrence $[l_1, PI_1]$ in the entire evolution. The dimension of this space is related to the Cartesian product of:

- all types of anti-patterns described in Section II (8),
- the number of non-inverted Allen relations (6) and
- the number of types of different spatial relations from Definition 3 (2).

Consequently, the occurrence of an anti-pattern is described by a vector of 96 natural numbers.

Please note that the notions of closeness and remoteness from Definition 3, as well as the notions of lifespan, occurrence and spatio-temporal relations, are defined in such a way that they are also applicable to any sub-graph that can be observed in snapshots of the software evolution. Thus, we can compute the aforementioned 96 attributes for any occurrence of such a subgraph.

In ([1]) we argued that such a vector appears to be very specific for occurrences of design anti-patterns. This phenomenon may be interpreted as a tendency for certain design anti-patterns to appear close to each other (spatial relation) and one after another (temporal relation). They can therefore be used to predict areas in the source code, where design anti-pattern may appear in the future. In this research we will use similar framework to improve detection quality for detectors described in Section II.

A. Spatio-temporal rules

We will describe a method of mining *spatio-temporal rules* which allows us to reason about spatio-temporal relations in the entire software evolution. We will construct these rules by applying a rule-based machine-learning classification algorithm on specially prepared decision table. The following paragraphs describe how this table is constructed.

In Section V we argued that for any occurrence of design anti-pattern and any occurrence of any other subgraph of software snapshot we can compute a vector of 96 attributes that describe its spatio-temporal relations. For each occurrence of design anti-pattern PI of type t (e.g. Blob) we will insert one row to the decision table, with 96 conditional attributes and decision= t . We will call this a positive row for t (e.g. positive row for Blob). To balance this we will pick a random subgraph that does not correspond to anti-pattern t and has the same number of nodes as PI and insert it into decision table with 96 conditional attributes computed likewise. For such a row we will set decision=NOT_ t (e.g. NOT_Blob). We will call such a row a negative row for t (e.g. negative row for Blob).

The decision table constructed according to the above description has 97 columns and the number of rows is twice the number of occurrences of all design anti-patterns in the entire software evolution. We can partition this table into smaller tables by selecting only positive and negative rows for only single type t of anti-patterns (e.g. we only take rows with decision Blob and NOT_Blob). Each such sub-table is in fact a perfectly balanced binary decision table that can be used to train a machine-learning classification algorithm that produces a classifier in the form of a set of classification rules. We will call these rules *spatio-temporal rules* for t and the classifier will be called *spatio-temporal classifier* for t . Technically, the classifier, given a vector of 96 natural numbers, outputs either t or NOT_ t .

If we take occurrence l of some subgraph g in the software evolution we can compute 96 attributes for it and apply a spatio-temporal classifier on such a vector. Conceptually, the classifier for type t can tell if the given graph occurrence resembles a design anti-pattern t occurrence in terms of its spatio-temporal relations. We can use this observation to introduce an improved spatio-temporal detector for t . This concept is described in the following section.

VI. SPATIO-TEMPORAL DETECTORS OF ANTI-PATTERNS

Let us assume that some subgraph g , that is part of a software snapshot at revision r , is considered to be an anti-pattern of type t by a respective detector described in Section II. If we have a spatio-temporal classifier for t , then we can find its output for the occurrence of g at revision r according to the method described in the preceding Section V-A. In the proposed approach we will consider g to be an actual anti-pattern of type t iff the output of the spatio-temporal classifier was also t . Conceptually, in this detection strategy we combine a purely static definition of design anti-patterns given in Section II with spatio-temporal knowledge about the evolution of the software. We will consider a graph to be an actual anti-pattern, only when both premises hold.

Clearly, such a compound classifier can reduce the number of false positives but also increase the number of false negatives, thus it does not necessarily improve the quality of an anti-pattern detection. However, in practice, it appears that such a construct improves the classification quality by an

average of 4% in terms of F-measure, if we mine the spatio-temporal rules from the very beginning of software evolution and use them to identify static patterns in a separate, final period of this evolution. Details of the experimental setting is given in the following Section VII.

VII. EXPERIMENTAL VALIDATION

The experiments were run on the evolution of the following open-source software:

- Argouml ([30], [31], [32]) is a simple UML editor, which used to be popular. The SCM of this software (along with Xerces2j and Jhotdraw) is frequently used as the source of data in mining software repositories research. The analyzed evolution of this software spans from January 1998 to December 2011.
- Struts1 ([33], [34], [35]) is a java web framework, which was popular 20 years ago. The analyzed evolution of this software spans from May 2000 to December 2008.
- Xerces2j ([36], [34], [35]) is a popular Java XML Parser. The analysed software evolution spans from November 1999 to May 2008.
- Elasticsearch ([37], [38]) is a popular search engine. The analyzed evolution of this software spans from February 2010 to September 2017.
- JHotdraw ([39], [40]) is a Java framework for 2D graphics. The analyzed evolution of this software spans from October 2000 to November 2012.
- Lucene-solr ([41], [42], [35]) is a popular search engine. The analyzed evolution of this software spans from September 2001 to November 2016.
- Wildfly ([43], [44], [45]) is a popular Java application server. The analyzed evolution of this software spans from June 2010 to June 2013.

For each system, the spatio-temporal classifier, based on C4.5 Boolean classifier, was trained on the sub-evolution built from the first 70% revisions of the respective system. The closeness and remoteness spatial relations were defined by $d_c = 1$ and $d_r = 2$ respectively (see Section III). The detection quality was tested on each commit of the sub-evolution which consisted of the last 30% of revisions. Table II shows the cases, where the result of detection was different. In two cases the quality decreased by 6-11%, and in all other cases it improved by 1-14% in terms of F1. There was an average improvement of 4%.

VIII. CONCLUSIONS

In our previous work ([1], [46]) we analyzed the phenomena of spatio-temporal relations between occurrences of anti-patterns in the software evolution. This paper presents how spatio-temporal rules can help to slightly improve the quality of detection of a few anti-patterns: We combine typical static detectors derived from existing state-of-the-art detection methods with additional knowledge that comes from analysis of the spatio-temporal relations in the software development process.

Dataset	APT	Spatial Prec./Rec.	Spatio-temp. Prec./Rec.	Change of F1
elastic	BB	0.71 / 0.88	1.0 / 0.54	0.89
argouml	Bl	0.9 / 0.76	1.0 / 0.76	1.05
elastic	Bl	0.83 / 0.9	1.0 / 0.88	1.08
Xerces	Bl	0.91 / 0.69	1.0 / 0.69	1.04
elastic	BC	0.87 / 0.84	1.0 / 0.81	1.05
argouml	SAK	0.78 / 1.0	1.0 / 0.94	1.11
elastic	SAK	0.78 / 0.99	1.0 / 0.99	1.14
Lucene	SAK	0.86 / 1.0	1.0 / 0.92	1.04
Xerces	SAK	0.8 / 0.89	1.0 / 0.65	0.94
Xerces	YoYo	0.98 / 1.0	1.0 / 0.99	1.01
			Average	1.04

Table II

IMPACT OF SPATIO-TEMPORAL RULES ON STATIC DETECTION QUALITY. THE TABLE PRESENTS HOW SPATIO-TEMPORAL RULES CHANGE THE QUALITY OF DETECTION OF SPATIAL DETECTORS DESCRIBED IN SUBSECTIONS II-C1–II-C8. THIRD COLUMN PRESENTS PRECISION/RECALL OF PURELY STATIC DETECTION. THE FOURTH COLUMN PRESENTS PRECISION/RECALL AFTER ADDING SPATIO-TEMPORAL RULES. APT = ANTI-PATTERN TYPE, BB = BASE BEAN, BL = BLOB, BC = BRAIN CLASS, SAK = SWISS ARMY KNIFE, F1 = F-MEASURE.

The experimental validation shows that in most cases the prediction quality was identical, with an observable difference only in a few cases described in Table II above. It was worse in only two cases, and on average in improved by 4% in terms of F-measure, which is a harmonic mean of precision and recall.

A. Future work

The following paragraphs provide some proposals for applications and modifications of the proposed framework, which yield to future research in the topic.

In this paper we have presented how spatio-temporal rules can be used to improve the quality of prediction of static pattern detection. The same rules can be used to predict where certain types of anti-patterns may appear in the future in the software source code.

In the method described herein, spatio-temporal rules were trained and used within the same software system. However, it is possible that rules trained on the evolution of one software system can be interpreted within another system. By doing so we may answer if there are universal spatio-temporal rules that model typical spatio-temporal phenomena in the software development process that hold across many projects.

The proposed framework is specifically suited for Java programming language, but can be easily adopted to other programming languages as well. It would require changing the definition of the software snapshot multigraph.

Allens algebra is a helpful formalism, but it can arguably be too simplifying when it is used to model temporal relations between intervals of revisions in the software development process. For example, it cannot measure temporal proximity between intervals. Please note that relation between the separated intervals of revisions are indiscernible in terms of Allens theory in two cases: when they are separated by a single commit and when they are separated by thousands of commits. Thus Allens algebra could be replaced by alternative formalism, which would incorporate more accurate model of temporal relations.

In this research we have assumed time to be linear (i.e. commits are linearly ordered), as we have considered commits from only a single main development branch. In fact, the software development process typically uses many parallel branches and cross-branch merges (see [47]). To cover such phenomena, the time representation in the proposed framework should be replaced with a more versatile model, such as e.g. CTL.

This research is based on the concept of a spatio-temporal relation to occurrences of anti-patterns described in Section II. But it can easily be adapted so that we can use other subgraphs in place of anti-patterns. For example we could use frequent subgraphs ([48]) or graphs built from frequently modified source code entities ([49]).

B. Threats to validity

Drawing general conclusions from empirical studies based on just a few software systems is always difficult, because of the complexity of the matter and the variety of different sources of data. This paper is based on data gathered from systems that share a common characteristic: they are open-source, non-commercial systems, developed by the community for many years. It may appear that software developed differently (e.g. by smaller teams, commercially, with closed source) tends to evolve differently.

Anti-patterns are very infrequent in relation to all possible containment-complete subgraphs in the software, as the number of the latter is exponential in relation to the number of software entities in the source code. To have a balanced set of examples, we have randomly selected sub-set of such subgraphs to construct a decision table described in Section V-A. Even though the results presented in Section VII were stable with multiple repetitions of the experimental reproduction, this fact presumably introduces some randomness in the results.

REFERENCES

- [1] Ł. Puławski, "Temporal Relations of Rough Anti-patterns in Software Development," in *Rough Sets*, ser. Lecture Notes in Computer Science, L. Polkowski, Y. Yao, P. Artiemjew, D. Ciucci, D. Liu, D. Ślęzak, and B. Zielosko, Eds. Cham: Springer International Publishing, 2017, pp. 447–464.
- [2] S. M. Olbrich, D. S. Cruzes, and D. I. K. Sjöberg, "Are all code smells harmful? A study of God Classes and Brain Classes in the evolution of three open source systems," in *Proceedings of the 2010 IEEE International Conference on Software Maintenance*, ser. ICSM '10. Washington, DC, USA: IEEE Computer Society, 2010, pp. 1–10.
- [3] W. H. Brown, R. C. Malveau, H. W. "Skip" McCormick, and T. J. Mowbray, *AntiPatterns: Refactoring Software, Architectures, and Projects in Crisis*, 1st ed. New York, NY, USA: John Wiley and Sons, Inc., 1998.
- [4] R. Marinescu, "Detection Strategies: Metrics-Based Rules for Detecting Design Flaws," *Software Maintenance, IEEE International Conference on*, vol. 0, pp. 350–359, 2004.
- [5] S. Olbrich, D. S. Cruzes, V. Basili, and N. Zazworka, "The evolution and impact of code smells: A case study of two open source systems," *Empirical Software Engineering and Measurement, International Symposium on*, vol. 0, pp. 390–400, 2009.
- [6] R. Wieman, *Anti-Pattern Scammer: An Approach to Detect Anti-Patterns and Design Violations*. LAP LAMBERT Academic Publishing, Nov. 2011.
- [7] H. Kagdi, M. L. Collard, and J. I. Maletic, "Towards a taxonomy of approaches for mining of source code repositories," *SIGSOFT Softw. Eng. Notes*, vol. 30, pp. 1–5, May 2005.

- [8] N. Moha, Y.-G. Guéhéneuc, L. Duchien, and A.-F. Le Meur, "DECOR: A Method for the Specification and Detection of Code and Design Smells," *Software Engineering, IEEE Transactions on*, vol. 36, no. 1, pp. 20–36, Jan. 2010.
- [9] N. Moha, Y.-g. Gueheneuc, and P. Leduc, "Automatic Generation of Detection Algorithms for Design Defects," in *21st IEEE/ACM International Conference on Automated Software Engineering (ASE'06)*. Tokyo: IEEE, 2006, pp. 297–300.
- [10] D. H. Taenzer, M. Ganti, and S. Podar, "Problems in Object-Oriented Software Reuse," in *ECOOP '89: Proceedings of the Third European Conference on Object-Oriented Programming, Nottingham, UK, July 10-14, 1989*, S. Cook, Ed. Cambridge University Press, 1989, pp. 25–38.
- [11] A. Stoianov and I. Sora, "Detecting patterns and antipatterns in software using Prolog rules," in *2010 International Joint Conference on Computational Cybernetics and Technical Informatics*. Timisoara: IEEE, May 2010, pp. 253–258.
- [12] F. Jaafar, Y. G. Gueheneuc, S. Hamel, and F. Khomh, "Mining the relationship between anti-patterns dependencies and fault-proneness," in *Reverse Engineering (WCRE), 2013 20th Working Conference On*. IEEE, Oct. 2013, pp. 351–360.
- [13] T. Zimmermann, "Changes and bugs Mining and predicting development activities," in *Software Maintenance, 2009. ICSM 2009. IEEE International Conference On*. IEEE, 2009, pp. 443–446.
- [14] C. Izurieta and J. M. Bieman, "A multiple case study of design pattern decay, grime, and rot in evolving software systems," *Software Quality Journal*, pp. 1–35, Feb. 2012.
- [15] M. Fowler and K. Beck, *Refactoring Improving the Design of Existing Code*, 1st ed. Addison-Wesley, Jul. 2013.
- [16] H. Li and W. Cheung, "An Empirical Study of Software Metrics," *IEEE Transactions on Software Engineering*, vol. SE-13, no. 6, pp. 697–708, Jun. 1987.
- [17] T. J. McCabe, "A complexity measure," in *Proceedings of the 2nd International Conference on Software Engineering*, ser. ICSE '76. San Francisco, California, United States: IEEE Computer Society Press, 1976, pp. 407+.
- [18] B. A. Nejmeh, "NPATh: A measure of execution path complexity and its applications," *Commun. ACM*, vol. 31, no. 2, pp. 188–200, Feb. 1988.
- [19] S. R. Chidamber and C. F. Kemerer, "A Metrics Suite for Object Oriented Design," *IEEE Trans. Softw. Eng.*, vol. 20, no. 6, pp. 476–493, Jun. 1994.
- [20] M. Hitz and B. Montazeri, "Measuring coupling and cohesion in object-oriented systems," in *Proceedings of International Symposium on Applied Corporate Computing*, 1995, pp. 25–27.
- [21] J. M. Bieman and B.-K. Kang, "Cohesion and reuse in an object-oriented system," *SIGSOFT Softw. Eng. Notes*, vol. 20, no. SI, pp. 259–262, Aug. 1995.
- [22] N. Moha, Y. G. Gueheneuc, A. F. Le Meur, L. Duchien, and A. Tiberghien, "From a domain analysis to the specification and detection of code and design smells," *Form. Asp. Comput.*, vol. 22, pp. 345–361, May 2010.
- [23] M. Fowler, *Patterns of Enterprise Application Architecture*, 1st ed. Addison-Wesley Professional, Nov. 2002.
- [24] D. Ratiu, S. Ducasse, T. Girba, and R. Marinescu, "Using History Information to Improve Design Flaws Detection," in *Proceedings of the Eighth Euromicro Working Conference on Software Maintenance and Reengineering (CSMR'04)*, ser. CSMR '04. Washington, DC, USA: IEEE Computer Society, 2004.
- [25] J. Din, A. B. Al-Badareen, and Y. Y. Jusoh, "Antipatterns detection approaches in Object-Oriented Design: A literature review," in *2012 7th International Conference on Computing and Convergence Technology (ICCT)*. IEEE, 2012, pp. 926–931.
- [26] F. Palomba, R. Oliveto, and A. De Lucia, "Investigating code smell co-occurrences using association rule learning: A replicated study," in *2017 IEEE Workshop on Machine Learning Techniques for Software Quality Evaluation (MaLTeSQuE)*, Feb. 2017, pp. 8–13.
- [27] F. Palomba, G. Bavota, M. Di Penta, R. Oliveto, D. Poshyvanyk, and A. De Lucia, "Mining version histories for detecting code smells," *IEEE Transactions on Software Engineering*, vol. 41, no. 5, pp. 462–489, 2014.
- [28] M. Zhang, N. Baddoo, P. Wernick, and T. Hall, "Improving the Precision of Fowler's Definitions of Bad Smells," in *2008 32nd Annual IEEE Software Engineering Workshop*. Kassandra, Greece: IEEE, Oct. 2008, pp. 161–166.
- [29] J. F. Allen, "Maintaining Knowledge About Temporal Intervals," *Commun. ACM*, vol. 26, no. 11, pp. 832–843, Nov. 1983.
- [30] (2015, Sep) Argouml. [Online]. Available: <http://argouml.tigris.org/issues/>
- [31] (2020, Nov) Argouml source code. [Online]. Available: <https://github.com/argouml-tigris-org/argouml>
- [32] (2020, Nov) Argouml source code. [Online]. Available: <http://argouml.tigris.org/servlets/ProjectIssues>
- [33] (2020, Nov) Struts. [Online]. Available: <https://struts.apache.org/>
- [34] (2020, Nov) Apache software foundation scm. [Online]. Available: <http://svn.apache.org/repos/>
- [35] (2020, Nov) Apache software foundation issue tracker. [Online]. Available: <https://issues.apache.org>
- [36] (2020, Nov) Eclipse jdt. [Online]. Available: <https://xerces.apache.org/#xerces2-j>
- [37] (2020, Nov) Elasticsearch. [Online]. Available: <https://www.elastic.co/>
- [38] (2020, Nov) Elasticsearch source code. [Online]. Available: <https://github.com/elastic/elasticsearch>
- [39] (2018, Apr) Jhotdraw. [Online]. Available: <http://www.jhotdraw.org/>
- [40] (2020, Nov) Jhotdraw source code. [Online]. Available: <https://github.com/wrandelshofer/jhotdraw>
- [41] (2020, Nov) Lucene solr. [Online]. Available: <https://solr.apache.org/>
- [42] (2020, Nov) Lucene solr source code. [Online]. Available: <https://gitbox.apache.org/repos/asf/lucene-solr.git>
- [43] (2020, Nov) Wildfly. [Online]. Available: <https://www.wildfly.org/>
- [44] (2020, Nov) Wildfly scm. [Online]. Available: <https://github.com/wildfly/wildfly>
- [45] (2020, Nov) Wildfly issue tracker. [Online]. Available: <https://issues.jboss.org>
- [46] L. Pulawski, "An automatic approach for detecting early indicators of design anti-patterns," in *JCKBSE*, ser. Frontiers in Artificial Intelligence and Applications, M. Virvou and S. Matsuura, Eds., vol. 240. IOS Press, 2012, pp. 161–170.
- [47] V. Driessen. (2010, Jan) <https://datasift.github.io/gitflow/IntroducingGitFlow.html>. [Online]. Available: <https://datasift.github.io/gitflow/IntroducingGitFlow.html>
- [48] C. C. Aggarwal and H. Wang, *Managing and Mining Graph Data*, 1st ed. Springer Publishing Company, Incorporated, 2010.
- [49] N. Nagappan and T. Ball, "Use of Relative Code Churn Measures to Predict System Defect Density," in *Proceedings of the 27th International Conference on Software Engineering*, ser. ICSE '05. St. Louis, MO, USA: ACM, 2005, pp. 284–292.

15th International Symposium on Multimedia Applications and Processing

SFTWARE Engineering Department, Faculty of Automation, Computers and Electronics, University of Craiova, Romania “Multimedia Applications Development” Research Centre

BACKGROUND AND GOALS

Multimedia and information have become ubiquitous on the web and communication services, creating new challenges for detection, recognition, indexing, access, search, retrieval, automated understanding, processing and generation of several applications which are using image, signal or various multimedia technologies.

Recent advances in pervasive computers, networks, telecommunications, and information technology, along with the proliferation of multimedia mobile devices—such as laptops, iPods, personal digital assistants (PDA), and smartphones—have stimulated the rapid development of intelligent applications. These key technologies by using Virtual Reality, Augmented Reality and Computational Intelligence are creating a recent multimedia revolution which will have significant impact across a wide spectrum of consumer, business, healthcare, educational and governmental domains. Yet many challenges remain.

We welcome papers covering innovative applications, practical usage but also theoretical aspects of the above mentioned trends. The key objective of this session is to gather results from academia and industry partners working in all subfields of multimedia and language: content design, development, authoring and evaluation, systems/tools oriented research and development. We are also interested in looking at service architectures, protocols, and standards for multimedia communications—including middleware—along with the related security issues. Finally, we encourage submissions describing work on novel applications that exploit the unique set of advantages including home-networked entertainment and games. However, innovative contributions which don't exactly fit into these areas are also welcomed to this session.

The Multimedia Applications and Processing (MMAP) will provide an opportunity for researchers and professionals to discuss present and future challenges as well as potential collaboration for future progress in the field. The MMAP Symposium welcomes submissions of original papers concerning all aspects of multimedia domain ranging from concepts and theoretical developments to advanced technologies and innovative applications. MMAP invites original previously unpublished contributions that are not submitted concurrently to a journal or another conference. Papers acceptance and

publication will be judged based on their relevance to the symposium theme, clarity of presentation, originality and accuracy of results and proposed solutions.

CALL FOR PAPERS

MMAP 2022 is a major forum for researchers and practitioners from academia, industry, and government to present, discuss, and exchange ideas that address real-world problems with real-world solutions.

The MMAP 2022 Symposium welcomes submissions of original papers concerning all aspects of multimedia domain ranging from concepts and theoretical developments to advanced technologies and innovative applications. MMAP 2022 invites original previously unpublished contributions that are not submitted concurrently to a journal or another conference. Papers acceptance and publication will be judged based on their relevance to the symposium theme, clarity of presentation, originality and accuracy of results and proposed solutions.

TOPICS

- Audio, Image and Video Processing
- Animation, Virtual Reality, 3D and Stereo Imaging
- Big Data Science and Multimedia Systems
- Cloud Computing and Multimedia Applications
- Machine Learning, Fuzzy Systems, Neural Networks and Computational Intelligence for Information Retrieval in Multimedia Applications
- Data Mining, Warehousing and Knowledge Extraction
- Multimedia File Systems and Databases: Indexing, Recognition and Retrieval
- Multimedia in Internet and Web Based Systems
- E-Learning, E-Commerce and E-Society Applications
- Human Computer Interaction and Interfaces in Multimedia Applications
- Multimedia in Medical Applications and Computational biology
- Entertainment, Personalized Systems and Games
- Security in Multimedia Applications: Authentication and Watermarking
- Distributed Multimedia Systems
- Network and Operating System Support for Multimedia
- Mobile Network Architecture and Fuzzy Logic Systems
- Intelligent Multimedia Network Applications
- Future Trends in Computing System Technologies and Applications
- Trends in Processing Multimedia Information

- Multimedia Ontology and Perception for Multimedia Users

BEST PAPER AWARD

A best paper award will be made for work of high quality presented at the MMAP Symposium. Award comprises a certificate for the authors and will be announced on time of conference. Selected papers will be invited to high IF journals organized for the participants of MMAP.

- Authors should submit draft papers (as Postscript, PDF or MSWord file).
- The total length of a paper should not exceed 10 pages IEEE style (including tables, figures and references). IEEE style templates are available here.
- Papers will be refereed and accepted on the basis of their scientific merit and relevance to the workshop.
- Preprints containing accepted papers will be published on a USB memory stick provided to the FedCSIS participants.
- Only papers presented at the conference will be published in Conference Proceedings and submitted for inclusion in the IEEE Xplore@database.
- Conference proceedings will be published in a volume with ISBN, ISSN and DOI numbers and posted at the conference WWW site.
- Conference proceedings will be submitted for indexation according to information here.
- Extended versions of selected papers presented during the conference will be published as Special Issue(s).
- Organizers reserve right to move accepted papers between FedCSIS events.

WINNERS OF MMAP 2019 BEST PAPER AWARD

- Depth Map Improvements for Stereo-based Depth Cameras on Drones. Authors: Daniel Pohl (Intel Corporation), Sergey Dorodnicov (Intel Corporation).
- Information theoretical secure key sharing protocol for noiseless public constant parameter channels without cryptographic assumptions. Authors: Valery Korzhik, Vladimir Starostin, Muaed Kabardov, Aleksandr Gerasimovich, Victor Yakovlev, Aleksey Zhuvikin (The Bonch-Bruевич Saint-Petersburg State University of Telecommunications), Guillermo Morales-Luna (Computer Science Department CINVESTAV-IPIV, Mexico City, Mexico).

ADVISORY BOARD

- **Neustein, Amy**, Boston University, USA
- **Jain, Lakhmi C.**, University of South Australia and University of Canberra, Australia
- **Zurada, Jacek**, University of Louisville, United States
- **Ioannis, Pitas**, University of Thessaloniki, Greece
- **Badica, Costin**, University of Craiova, Romania
- **Borko, Furht**, Florida Atlantic University, USA
- **Kosch, Harald**, University of Passau, Germany
- **Uskov, Vladimir**, Bradley University, USA
- **Deserno, Thomas M.**, Aachen University, Germany
- **Burdescu, Dumitru Dan**, University of Craiova, Romania

TECHNICAL SESSION CHAIR

- **Schiopoiu Burlea, Adriana**, University of Craiova, Romania

A Modified ICP Algorithm Based on FAST and Optical Flow for 3D Registration

Konrad Koniarski
Systems Research Institute
Polish Academy of Sciences
ul. Newelska 6, 01-447 Warsaw, Poland
Email: konrad.koniarski@gmail.com

Andrzej Myśliński
Systems Research Institute
Polish Academy of Sciences
ul. Newelska 6, 01-447 Warsaw, Poland
Email: andrzej.myslinski@ibspan.waw.pl

Abstract—This paper presents a modified Iterative Closest Point (ICP) algorithm based on a suitable selection of initial points and local optical flow to speed up registration of static scenes with high accuracy. The biggest disadvantages of using standard ICP algorithm are appropriate initialization and effective matching point step in each iteration. In the proposed modification we deal with these problems and optimize this method for Augmented Reality application. As this application uses RGB-D images sequence the changes between consecutive key-frames are small. Therefore only small subset of the source image key-points is selected using scale-space pyramid and FAST approaches. It leads to the significant reduction of the number of the processed image points. Since the point matching technique using local optical flow is applied, in each optimization step of ICP the costly point matching procedure can be abandoned. The proposed approach has been validated by the numerical examples.

I. INTRODUCTION

THE reconstruction of the geometry of the environment from a movable camera is a well studied problem in the category of computer vision topics. In theory, the determination of the trajectory is already possible with the use of only a few reference points traced in real time [1]. However, the smaller number of tracked points makes the solution more sensitive to data noise. Recently, we have seen significant progress in the development of methods for dense 3D reconstruction from many images. Unfortunately, many of these proposed approaches are not able to work in real time [2]. In addition, they usually require a large number of calibrated images, making them unsuitable for live reconstruction from one movable camera. On the other hand, there are many approaches for the reconstruction of dense depth maps from pairs of images [3]. While these approaches have been shown to provide excellent results in dense depth estimation, they are usually computationally too expensive for real-time applications.

In this article we propose modification of the first block of Iterative Closest Point (ICP) method - looking for a match of the identities of the points in each iteration. It has been replaced with the previous step using local optical tracking. This approach, perhaps surprisingly, does not involve a significant loss of speed, but allows a significant reduction in the amount of data while increasing accuracy. The modified ICP method was designed for Augmented Reality (AR) applications. AR

is the computer vision system integrating computer generated virtual information with the real world environment in the form of image, sound or video. The added information, usually virtual objects, has to be precisely aligned with the real world. Image registration is the key element of AR algorithm. The ICP method was first presented by Besl and Mackay in 1992 (see [4]). This method is more concept then solid algorithm. The first step, i.e., the initial selection of the points is the most important step to enforce ICP method to be convergent to global rather than to local minimum. There is always a trade off between using feature points or dense data. In literature ICP approach is based on feature extracting methods as in [5]. It leads to long computational time and low quality of point cloud to be augmented in final step of AR image generation. Therefore in this paper, we present different approach than proposed in literature for initial points selection and matching based on scale-space and Features from Accelerated Segment Test (FAST) [6], [7] approaches as well as local optical flow method. The selected set of points may contain outliers. However, in the proposed method there is no separate mechanism for removing such points. Outliers are removed in the optical tracking procedure if they do not meet the stability criteria of this procedure. As a result, we get a set of points that no longer contain outliers. Finally in the last step the error metric is minimized to find the six parameters of the transformation, i.e. the rotation matrix and the translation vector. The main focus is on the speed of convergence and the accuracy of the final transformation and the application it to construct AR image. The performance of the ICP method depends mainly on the data and proper initialization. If some a-priori assumption concerning the similarity of the frames can be made, the quality of matching is much better. Therefore in this paper, we assume that point clouds are constructed from two following frames of a video sequence. The pixel brightness is also assumed not to change significantly on those two consecutive frames.

II. RELATED WORK

A. Geometric Registration

Most of the registration methods operate on candidate correspondences. Popular method use point to point matches

based on local geometric descriptors [8], other defines correspondence on pairs or tuples of points [9]. When candidate correspondences are collected, alignment is estimated attractively from sparse subset to correspondence. This iterative process is typically based on variant of randomized algorithms like RANSAC [8], [9]. When the data is noisy and the surfaces only partially overlap, existing pipelines often require many iterations to sample a good correspondence set and find a good reasonable alignment. In many applications we have a priori knowledge that stream of consecutive data observation has relatively small transformations. This approach is known in literature as local refinement where rough initial alignment is known and the result is tight registration usually based on denser correspondence compared to global alignment [10]. ICP method and its modifications are popular for local refinement. The simplest algorithm of the ICP starts with initial alignment and alternates between establishing correspondence via find the closest point and recalculate the alignment based on the current correspondence set. ICP can give an accurate result when initiated near the optimal position, but it is unreliable without such initialization. In many papers [11], [12] are explored different approaches to increase ICP sensitivity to local optima. There are many modification of ICP method based on correspondence or transformation parameter estimation step modifications [5]. Park [11] proposed modification where he used geometry as well as intensity of RGB color value. This approach is valid when registration use data stream from the source when conditions are not changed like frame stream where camera intrinsic parameters are the same. The accumulated 3D model can be either in the form of a volumetric representation [12], a 3D point cloud [13] or a set of depth maps.

B. Optical flow of feature points

Optical flow is popular method of image processing when there is a need to know the movement (speed and direction) of the part of the image. According to Akpınar et al. [14], optical flow estimation algorithms can be grouped according to the theoretical approach while interpreting optical flow. These are differential techniques, region-based matching, energy-based methods and phase-based techniques. There are two groups of optical flow methods. The first is local optical flow introduced by Lucas-Kanade [15], and the second global optical flow introduced by Horn and Schunck [16]. In this paper we mostly focus on geometric aspect of registration then local optical flow is more suitable [7], [17]. Input data stream consists color RGB intensity frames and depth information.

III. PROPOSED ALGORITHM

A. Overview

Our goal is to calculate the rigid body motion transition T consisting of translation t and rotation R that minimizes the difference between the two sets of points O and M . In the next subsections, the process of locating, matching, and filtering the appropriate feature points is described and the following section presents the proposed pose estimation calculation.

B. Finding and Matching Visual Features

An RGB-D image consists of a color image I and a depth image D recorded in the same coordinate frame. We assume to have a pair of RGB-D (I_i, D_i) and (I_j, D_j) images and an initial T^0 transformation that roughly aligns (I_i, D_i) to (I_j, D_j) . Also $p = (u, v)^T$ is the pixel in (I_i, D_i) and $p' = (u', v')^T$ is the corresponding pixel in (I_j, D_j) . The goal is to find the optimal transformation that densely aligns the two RGB-D images. Here we assume that the individual frames are not distant in time, so that the color intensity of the pixels does not change rapidly and the initial match T^0 can be equal to the identity matrix. The first step of registration is to select feature points using FAST method on RGB image. For image I FAST point detecting method gives set of feature points P . In order to avoid the problem with the scale and the high speed of moving individual points, a scale pyramid is built. The original RGB image I is used in first layer, then the size of image is divided by 2 and the resultant image, the same procedure is repeated until the image becomes too small. As a result the set $N = \{P_k\}$ of points is obtained where k is number of layers in pyramid. Then local optical flow method is used for tracking of selected points $N_{i \rightarrow j}$ from image I_i on consecutive image I_j . Let $O_{i \rightarrow j}$ be the set of successfully tracked points and $C_{i \rightarrow j}$ their movement vectors. Then the projections π of the RGB-D image pixel $p = (u, v)^T$, $d = D(p)$ over 3D space is done using

$$\Pi(u, v, d) = \left[\frac{(u - c_x)d}{f_x}, \frac{(v - c_y)d}{f_y}, d, 1 \right]^T, \quad (1)$$

where f_x and f_y are the camera focal lengths and (c_x, c_y) is the principal point. The inverse projection function π^{-1} is defined as follow

$$\pi^{-1}(x, y, z, 1) = \left(\frac{xf_x}{z} + c_x, \frac{yf_y}{z} + c_y, z \right)^T. \quad (2)$$

Using the projection π over points set $O_{i \rightarrow j}$ the 3D point set is obtained.

$$W_{i \rightarrow j} = \pi(O_{i \rightarrow j}) \quad (3)$$

In practice, the depth component in an RGB-D image need not always be defined. Then such a pixel cannot be projected into 3D space. In that case such pixels are not used for 3D projection. Set $W_{i,j}$ is also called point cloud. The photometric objective is to find transformation T satisfying:

$$p = \pi^{-1}(T\pi(p', D(p'))). \quad (4)$$

Projection of pixels p and p' should be the same point in 3D space

$$\pi(p, D(p)) = T\pi(p', D(p')). \quad (5)$$

C. Pose estimation

The ICP consist of two steps correspondence estimation and transformation parameter estimation. In the first iteration we consider two consecutive image frames: current (I_i, D_i) and

registered (I_j, D_j) . The objective of correspondence estimation step of ICP is to build mapping function ϕ which define correspondence between I_i and I_j

$$p = \phi(p'). \quad (6)$$

In the proposed method function ϕ is defined by $C_{i \rightarrow j}$ and it does not have to be recalculated at each loop step. The transformation parameters estimation is done by looking for transformation T that minimize objective function

$$\begin{aligned} T_{n+1} &= \operatorname{argmin}_T \sum_{i=1}^O \|\pi(p_i, D(p_i)) - \\ &\quad T_n \pi(p'_i, D(p'_i))\|^2 \\ &= \operatorname{argmin}_T \sum_{i=1}^O \|\pi(\phi(p'_i), D(\phi(p'_i))) - \\ &\quad T_n \pi(p'_i, D(p'_i))\|^2, p_i, p'_i \in O \end{aligned} \quad (7)$$

where n is iteration step counter. The computations may be completed when the predetermined number of iterations have been performed or when the estimate of T is sufficient. Optimization problem (7) is solved using Gauss-Newton method. In each iteration, we linearize T locally as a 6 elements vector $\xi = (\omega_1, \omega_2, \omega_3, t_1, t_2, t_3)$. ξ contains rotation component ω and a translation component t .

$$T \approx \begin{pmatrix} 1 & -\omega_3 & \omega_2 & t_1 \\ \omega_3 & 1 & -\omega_1 & t_2 \\ -\omega_2 & \omega_1 & 1 & t_3 \\ 0 & 0 & 0 & 1 \end{pmatrix} T^n \quad (8)$$

Using the Gauss-Newton optimization scheme, we calculate ξ by solving the linear system

$$J_r^T J_r \xi = -J_r^T r \quad (9)$$

where r is the residual vector and J_r is Jacobian. In each optimization step n , r and J_r are calculated. Then T_n is obtained from ξ (see equation 8). Next T is updated by T_n using transformation to $SE(3)$ group.

IV. NUMERICAL EXPERIMENTS

Publicly available sequenced RGB-D framesets were used for the qualitative evaluation of the method proposed in this paper. All datasets used for test were published by Sturm [10]. RGB-D frames were registered using Microsoft Kinect device. Along with the images, the proposed benchmark dataset also provides the real trajectory taken by the camera acquired by an external, high-precision motion interception system.

A. Performance comparison

The performance comparison is summarized in the Table I. Processing time and error were calculated for different RGB-D sequences. Processing time was calculated as average of the time for registration consecutive frames. Mean error was calculated as the difference between benchmark trajectory and method calculated trajectory. Proposed method was compared to standard ICP implementation.

Algorithm 1 RGB-D images alignment

Require: Pair of RGB-D images $(I_i, D_i), (I_j, D_j)$, initial transformation T^0

Ensure: T registration transformation from frame i to j
 Build scale pyramid for RGB images I_i, I_j
 Calculate feature points using FAST method $N = \{P_k\}$
 Calculate local optical flow for points N and get $O_{i \rightarrow j}, C_{i \rightarrow j}$
 Project N into 3D space using projection equation (1)
while not converged **do**
 $r \leftarrow 0, J_r \leftarrow 0$
 Use $C_{i \rightarrow j}$ as correspondence between points from frames i and j
 Solve equation (9) to get ξ
 Update T using equation (8) and map to $SE(3)$
end while

TABLE I: Speed comparison of proposed method and ICP

Dataset	Proposed method		ICP method	
	Processing time [ms]	Mean error [m]	Processing time [ms]	Mean error [m]
fr1/xyz	0.123	0.251	0.182	0.511
fr1/rpy	0.128	0.262	0.195	0.25
fr2/xyz	0.131	0.17	0.152	0.19
fr2/rpy	0.190	0.291	0.211	0.31

B. Scene reconstruction

Single frame reconstruction is presented on Figure 1. Top row presents RGB image and depth component that is used for projection. Bottom row contains 3D projections of frame and camera location related to scene.

C. Augmented Reality application

Proposed method was used to build AR system as example of usage. Image 2 presents RGB frames and model position for different views as well as final AR image. In this case, the model image in the appropriate position is presented on the RGB image frame. The problem of obscuring an object added by scene elements is not considered.

V. CONCLUSION

The article presents an approach to the problem of image registration in a multi-frame sequence. The proposed method uses information obtained by an RGB-D camera moving in a static environment and is compared to the ICP method, another popular approach often used in similar applications. Its performance was assessed in terms of accuracy and processing time on the benchmark data sets. The proposed method achieved an average accuracy of 12% better than ICP and the processing time on average 37% better than ICP. The proposed algorithm works for both consecutive images and multiple image frames. The novelty of the use of local optical flow allows for better results than in the case of the classic ICP algorithm. The application of the proposed method has been shown on the example of AR. Information from previous frames is accumulated in the form of a point cloud. This

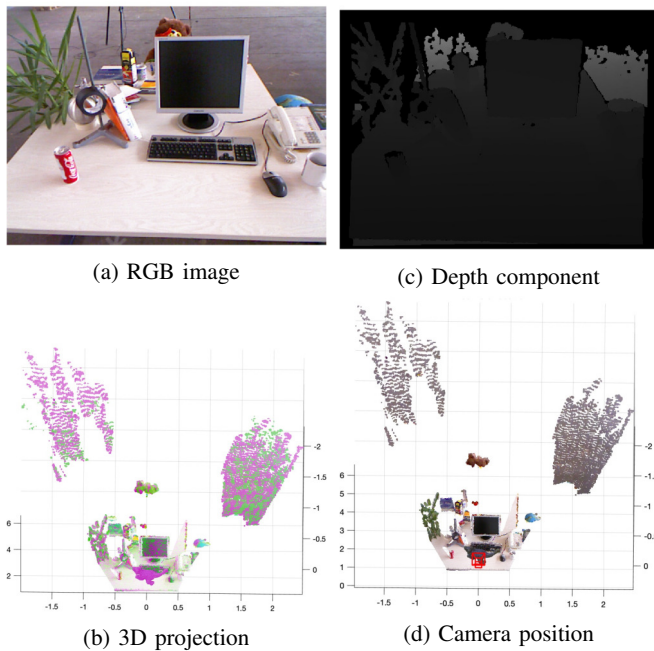


Fig. 1: Single frame projection into 3D scene.

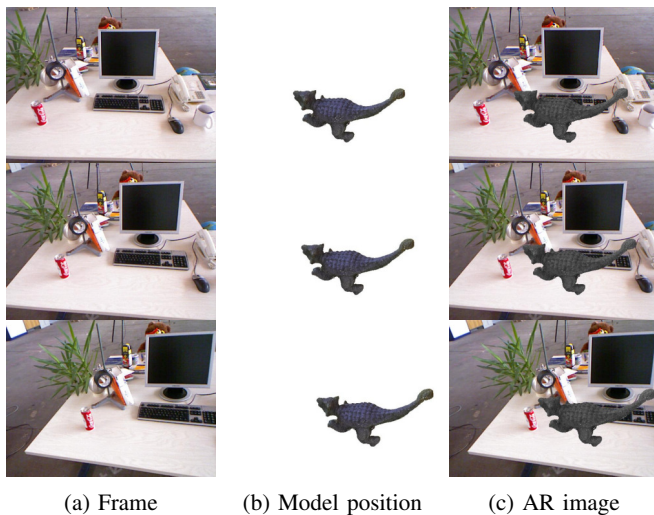


Fig. 2: AR system based on proposed registration method.

feature can be used in applications where 3D reconstruction plays an important role. In this article we not deal with the loop closure problem. However this method potentially could be used for simultaneous location and mapping algorithm (SLAM) application as well. This will be studied in future work.

REFERENCES

- [1] Q. Y. Zhou, J. Park, and V. Koltun, "Fast global registration," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9906 LNCS. Springer Verlag, 2016. doi: 10.1007/978-3-319-46475-6_47. ISBN 9783319464749. ISSN 16113349 pp. 766–782.

- [2] A. W. Fitzgibbon, "Robust registration of 2D and 3D point sets," *Image and Vision Computing*, vol. 21, no. 13-14, pp. 1145–1153, 12 2003. doi: 10.1016/J.IMAVIS.2003.09.004
- [3] C. Kerl, J. Sturm, and D. Cremers, "Dense visual SLAM for RGB-D cameras," in *IEEE International Conference on Intelligent Robots and Systems*, 2013. doi: 10.1109/IROS.2013.6696650. ISBN 9781467363587. ISSN 21530858 pp. 2100–2106.
- [4] Y. He, B. Liang, J. Yang, S. Li, and J. He, "An Iterative Closest Points Algorithm for Registration of 3D Laser Scanner Point Clouds with Geometric Features," *Sensors* 2017, Vol. 17, Page 1862, vol. 17, no. 8, p. 1862, 8 2017. doi: 10.3390/S17081862. [Online]. Available: [https://www.mdpi.com/1424-8220/17/8/1862](https://www.mdpi.com/1424-8220/17/8/1862/html)<https://www.mdpi.com/1424-8220/17/8/1862>
- [5] E. Marchand, H. Uchiyama, and F. Spindler, "Pose Estimation for Augmented Reality: A Hands-On Survey," *IEEE Transactions on Visualization and Computer Graphics*, vol. 22, no. 12, pp. 2633–2651, 12 2016. doi: 10.1109/TVCG.2015.2513408
- [6] E. Rosten and T. Drummond, "Fusing points and lines for high performance tracking," *Proceedings of the IEEE International Conference on Computer Vision*, vol. II, pp. 1508–1515, 2005. doi: 10.1109/ICCV.2005.104
- [7] Koniarski, "Augmented reality using optical flow," *Proceedings of the 2015 Federated Conference on Computer Science and Information Systems, FedCSIS 2015*, pp. 841–847, 10 2015. doi: 10.15439/2015F202. [Online]. Available: <https://fedcsis.org/proceedings/2015/2015F202.html>
- [8] Mahesh and M. V. Subramanyam, "Automatic feature based image registration using SIFT algorithm," in *2012 3rd International Conference on Computing, Communication and Networking Technologies, ICCCNT 2012*, 2012. doi: 10.1109/ICCCNT.2012.6396024
- [9] A. Fontes and J. E. B. Maia, "Visual Odometry for RGB-D Cameras," 3 2022. doi: 10.48550/arxiv.2203.15119. [Online]. Available: <https://arxiv.org/abs/2203.15119v1>
- [10] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *IEEE International Conference on Intelligent Robots and Systems*, 2012. doi: 10.1109/IROS.2012.6385773. ISBN 9781467317375. ISSN 21530858 pp. 573–580.
- [11] J. Park, Q. Y. Zhou, and V. Koltun, "Colored Point Cloud Registration Revisited," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2017-October, pp. 143–152, 12 2017. doi: 10.1109/ICCV.2017.25
- [12] R. A. Newcombe, R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon, "KinectFusion: Real-time dense surface mapping and tracking," in *2011 10th IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2011*, 2011. doi: 10.1109/ISMAR.2011.6092378. ISBN 9781457721830 pp. 127–136. [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.221.100>
- [13] K. Koniarski and A. Myśliński, "Feature Point Cloud Based Registration in Augmented Reality," *Lecture Notes in Networks and Systems*, vol. 364 LNNS, pp. 418–427, 12 2021. doi: 10.1007/978-3-030-92604-5_37. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-030-92604-5_37
- [14] S. Akpinar and F. N. Alpaslan, "Optical flow-based representation for video action detection," *Emerging Trends in Image Processing, Computer Vision and Pattern Recognition*, pp. 331–351, 1 2015. doi: 10.1016/B978-0-12-802045-6.00021-1
- [15] B. D. Lucas and T. Kanade, "Iterative Image Registration Technique With an Application to Stereo Vision," vol. 2, 1981, pp. 674–679. [Online]. Available: https://www.researchgate.net/publication/215458777_An_Iterative_Image_Registration_Technique_with_an_Application_to_Stereo_Vision_IJCAI
- [16] B. K. Horn and B. G. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, no. 1-3, pp. 185–203, 8 1981. doi: 10.1016/0004-3702(81)90024-2
- [17] A. Bruhn, J. Weickert, and C. Schnörr, "Lucas/Kanade meets Horn/Schunck: Combining local and global optic flow methods," *International Journal of Computer Vision*, vol. 61, no. 3, pp. 1–21, 2 2005. doi: 10.1023/B:VISI.0000045324.43199.43

Hierarchical data structures in rendering scenes containing a massive number of light sources

Andrzej Lamecki, Krzysztof Kaczmarek, Joanna Porter-Sobieraj
 Warsaw University of Technology, Faculty of Mathematics and Information Science
 ul. Koszykowa 75, 00-662 Warszawa, Poland
 Email: {andrzej.lamecki.stud, krzysztof.kaczmarek, joanna.porter}@pw.edu.pl

Abstract—In order to speed up the process of rendering scenes containing many light sources, spatial data structures are used, which allow the number of lights processed for each pixel to be reduced during lighting computation. Examples of algorithms using such data structures are clustered shading and hybrid lighting. Alongside the rendering time, it is important to consider memory consumption resulting from processing a large number of lights. This paper presents a novel modification of the hybrid lighting algorithm using an octree that allows for a significant reduction in the amount of memory required to store the data structure.

The proposed modification uses an octree to store the information about the rendered space. Detailed analysis of the proposed algorithm, and numerical results obtained for various 3D scenes, as well as different input data, all prove that the proposed method significantly reduces the memory required to store lists of lights used by the algorithm.

I. INTRODUCTION

IN RECENT years during the creation of virtual scenes, a lot of emphasis has been put on rendering visually realistic scenes. Rendering scenes containing multiple light sources requires calculating for each pixel a list of lights that affect its color. The most commonly used type of light is a point light with a limited range. Such lights can be represented as spheres in a rendered scene. The process of rendering a scene containing multiple light sources is therefore equivalent to determining for each rendered point which spheres contain this point. Fig. 1 shows an example scene for this problem containing 1 000 000 lights.

In the case of scenes with a large number of lights a naive approach of checking the distance between each rendered point and each light source requires a large number of operations. Improving rendering performance can be achieved by parallelization and by using spatial data structures to approximate light distribution in the scene space. These data structures are used during lighting computations to reduce the number of lights that are processed for each rendered point, which results in lower rendering times.

Another major concern in the rendering process, alongside the number of performed operations, is the memory complexity of the algorithm. Operating with a large amount of memory can often create a bottleneck while processing and visualizing complex scenes.

Research funded by the grant of Faculty of Mathematics and Information Science no. 504/04628/1120, Warsaw University of Technology

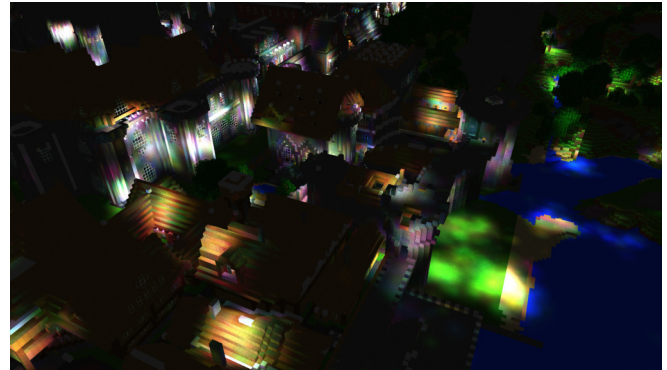


Fig. 1. A fragment of the *Rungholt* scene [1] containing 1 000 000 lights. All lights are point lights with limited range.

II. RELATED WORK

Research on the optimization of rendering has been conducted for over 50 years. Most of the algorithms used to efficiently render scenes with multiple lights use an additional data structure, describing scene geometry as well as lights placed in the scene, in order to decrease the time required for the calculation of lighting for each pixel. In these algorithms data structures allow a list of lights that potentially affect rendered geometry to be determined. The final decision on whether or not a light should affect a given pixel is made for each pair (pixel and light), based on the position of the shaded point and the position of the light source and, optionally, the normal vector in this point. The usage of the aforementioned data structures decreases the set of lights that are considered for a given pixel, which results in shorter rendering times.

One of the methods of rendering scenes containing many light sources is an algorithm using g-buffers [2], which are separate buffers used to store partial data, such as world position, normal vector or material properties, used in lighting computations in the final stage of frame rendering in which the final color displayed on the screen is calculated. Areas affected by lights are represented by spheres and each of those spheres is then rasterized onto the screen, and lighting data for each pixel in the area taken up by the light is updated. This solution is not efficient on modern GPUs [3] as rendering each frame requires multiple reads and writes to g-buffers, which significantly slows down the overall rendering process.

One of the most popular algorithms used to render scenes containing many light was the tiled shading algorithm [4], in which the rendered frame is divided into rectangular tiles (Fig. 2). Each tile is associated with a list of lights that affect any part of the scene contained in that tile. This additional data structure is used during final pixel color calculation. However, for each shaded pixel not all lights assigned to a corresponding list affect its color. Increasing the number of tiles can result in fewer lights to process for each pixel but processing more tiles requires additional operations and memory.

The part of the scene being rendered is divided only vertically and horizontally during tile construction. If a tile contains objects that are close to the camera and objects that are far from the camera (an example of such situation is presented in Fig. 3), lights affecting any of these objects will be processed for all of them. This can lead to many operations being performed unnecessarily as these lights are unlikely to affect all of the geometry contained in the tile.

There has been developed a modification of the tiled shading algorithm, a 2.5D culling [5], aiming to reduce the number of lights assigned to each tile in the case of a discontinuous geometry. For each tile, the geometry's range of depth is calculated and this range is divided equally into a fixed number of cells which contain the actual lists of lights.

An extension of the method of dividing space in three dimensions was proposed in the clustered shading algorithm [6]. In this algorithm all of the pixels of the rendered image are divided into groups (clusters) and a list of lights is assigned to each such cluster, in a similar way to the tiled shading algorithm. Clusters are created based on the three-dimensional position of shaded pixels as well as, optionally, a normal vector at that point. As described by Olsson et al. the depth of the rendered scene can be divided uniformly in either screen space or view space, or one can perform an exponential depth division in which resulting cells' dimensions are as equal as possible. During each frame of the rendering process, the lights are organized into a bounding volume hierarchy (BVH), a tree structure that allows for fast queries for all the lights that affect a given part of space. The tree is constructed in parallel, using the bottom-up approach. Then the bounding box of each cluster is used to determine a set of lights that possibly affect pixel samples in the cluster, and the normal vectors of cluster samples are used to discard lights that cannot affect any sample in the cluster.

There were also optimization attempts using the graphics pipeline to organize lights into lists assigned to different parts of the scene. The hybrid lighting algorithm [7] uses rectangular billboards to approximate the location of each light and to assign lights to appropriate lists. As with the clustered shading algorithm, the rendered space is divided into cells of a three-dimensional array. The billboards are rendered in a resolution corresponding to the vertical and horizontal array dimensions, and analytical calculations (the intersection of a sphere representing the area affected by the light and a ray originating from the camera) are performed to determine the range of cells in the depth affected by the light.

Complex spatial data structures can be implemented efficiently using graphics cards, allowing for parallel construction which results in lower building times. Tree structures are often used to organize points in space, e.g. for dealing with the level of detail [8] or multi-dimensional data clustering [9]. Trees allow the space to be divided into either regular cells [8], [9] or using hyper-planes which divide the space into two sub-spaces, each containing a subset of points [10].

The graphics pipeline has also been used to build this type of structure. Crassin et al. [11] describe a parallel algorithm for constructing an octree by inserting leaves into the partially constructed tree. To ensure that no two threads try to insert children nodes into the same parent node at the same time, a mutex for each node is used. The algorithm uses a separate buffer to store all nodes that cannot be inserted at a given moment and iterates until all of the nodes are inserted into the tree.

Another approach to rendering scenes with many light sources is taken in the tiled light trees algorithm [12]. Instead of the discrete division of the rendered space in all three dimensions, the lights are assigned to two-dimensional tiles, in a similar way to the tiled shading algorithm. However, in each tile a "light tree" (a variant of an interval tree), organizing lights in the depth of the scene, is constructed. During the final shading process for a given pixel, a tree from the tile in which the pixel is located is queried in the logarithmic time for the set of lights that can affect the pixel's color.

III. ALGORITHM DESCRIPTION

The developed algorithm is based on the hybrid lighting algorithm [7]. The algorithm's major novelty is its utilization of an octree for lights' spatial organization combined with dynamic analysis of the scene: tree leaves are only created in the presence of a scene's geometry. In this way, the memory needed for the tree's representation is minimized. Another optimization extends the basic tree properties: lights can be stored not only in the leaves but also in the internal nodes. This allows the information about a light to be stored in a single node if it is assigned to the lists of all its children nodes. The tree's creation is therefore faster, since there is just one insertion operation instead of many.

One of the improvements in the tree's construction is the adaptive space division during the tree's creation, instead of equal division in all directions. A bigger tree is created but only a fragment of it is used while the rest, laying outside of the divider region, is ignored. An example of this approach is presented in Fig. 4. The number of tree cells in each direction is equal to the lowest power of 2 equal or greater to the highest array dimension. This solution does not add any significant computation or memory cost due to the sparse structure of the octree used in the implementation.

The octree is constructed in each frame, before the light assignment operation. This operation is split into two steps: determining a list of cells containing the scene's geometry present in the rendered frame and building an octree containing these cells. In order to calculate a list of unique cells, first -

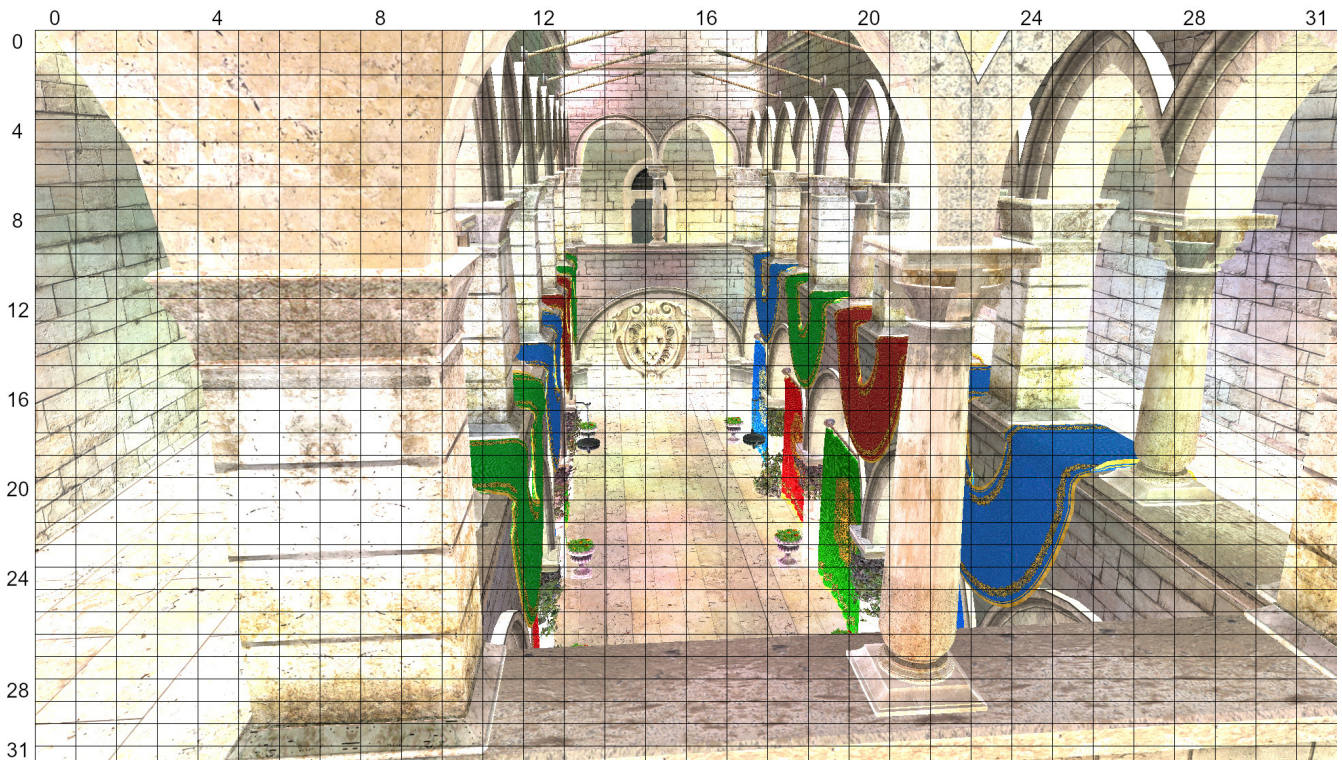


Fig. 2. Division of a rendered frame into 32 tiles vertically and 32 tiles horizontally (*Sponza* scene [1]).

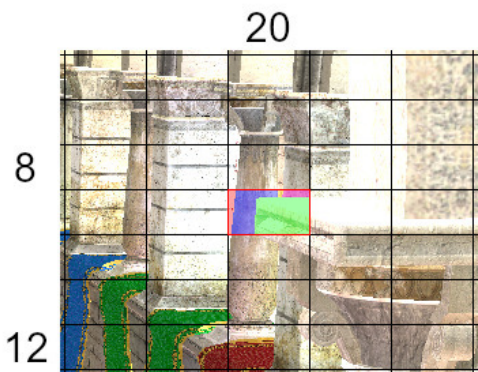


Fig. 3. Close-up of one of the tiles (outlined in red) of an image rendered using the tiled shading algorithm (*Sponza* scene [1]) in which there are many geometry discontinuities. Continuous parts of the geometry are highlighted in the same color.

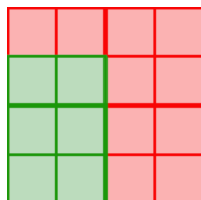


Fig. 4. Division of the space to 2×3 cells, based on 4×4 quadtree. Green cells represent a space fragment, red cells are outside and are ignored.

for each pixel of a rendered scene - the cell which the pixel belongs to is determined and its index is stored in a list. Then, repetitive values are removed from the list.

Two methods of removing repetitive elements have been proposed. The first method uses two functions commonly used in parallel computing: sorting and removing consecutive repeating elements on a list. One possible optimization of this process entails also removing consecutive repeating elements from the list before sorting. This optimization exploits the fact that all cell indices are written to the list row by row and many consecutive indices on the list are equal. This results in a certain amount of the list's elements being removed, which reduces the time needed to sort the entire list.

The second method of removing repeating elements from the list exploits a fact that all indices are from a small, finite range, bounded by the array dimensions. This makes it possible to use bitmasks to store the information about the presence of a cell in a frame. Moreover, the cells' indices are correlated with the two-dimensional tile of the image the pixel belongs to. This fact allows for easier parallel processing of the cells: all pixels from a given tile are processed by threads from the same warp, and the atomic operations (synchronized between threads in a single warp) are used to mark the presence of a given cell. The bitmasks are therefore stored in a separate list, each 32-bit element of the list representing 32 consecutive cells within the same two-dimensional tile. In the list containing the bitmasks, a parallel scan counting the

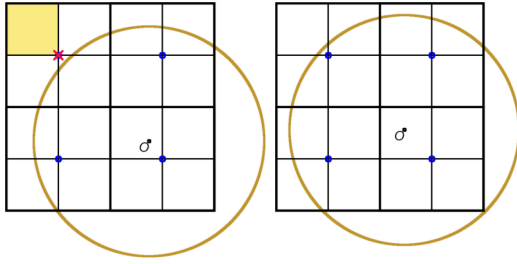


Fig. 5. The process of checking if a light can be assigned to the inner node of a quadtree. The node represents a group of 4×4 cells. The distance is checked from the light source O to each cell corner marked in blue. The light's range is represented by the yellow circle. In the example shown in the figure on the left, one of the points (marked in red) is farther from the light source than the light's range, therefore this light isn't assigned to the leaf marked in yellow so it cannot be assigned to the processed node. In the example shown in the figure on the right, all marked points are within range of the light, thus this light can be assigned to the processed node.

number of set bits is performed to determine the starting index in the resulting list under which the cells' indices should be written. In the final step, the list of unique cells present in the rendered frame is filled using the bitmask list.

The approach to building an octree described by Crassin et al. [11] has been adapted to the CUDA framework. All of the cells from the list computed in the previous step are inserted into the tree. This tree is then used during the assignment of lights to lists associated with each cell.

The process of assigning lights to lists is similar to the original algorithm, in which the lights' locations and ranges are approximated by billboards. The main difference is that after determining the cells which the light should be assigned to, for each cell a path in the octree (from the root to the leaf corresponding to the cell) is traced to check whether or not the light can be assigned to one of the internal nodes. For each node on this path, the distance from the light source to the innermost corners of the outermost cells of a group is compared to the light range. If all of the corners are within the range of the light, a light is assigned to this inner node and the rest of the path to the leaf is ignored. A 2D example of this process is shown in Fig. 5. All the calculations are performed in view-space because then the corner of each of the cells has a constant position. Moreover, it is possible to calculate the view-space position of each cell corner in advance and read it from the additional buffer instead of calculating it every time it is used.

During the final shading process the lights are read from all the lists corresponding to nodes on a path from the octree root to the leaf representing the cell the shaded pixel is in.

IV. RESULTS

The described algorithm was subjected to a series of tests to determine its effectiveness in comparison to the clustered shading and hybrid lighting algorithms. We concentrated on comparing the memory required by algorithms, as minimizing the memory requirements was the main purpose of the proposed modification. All reported results were obtained on a PC

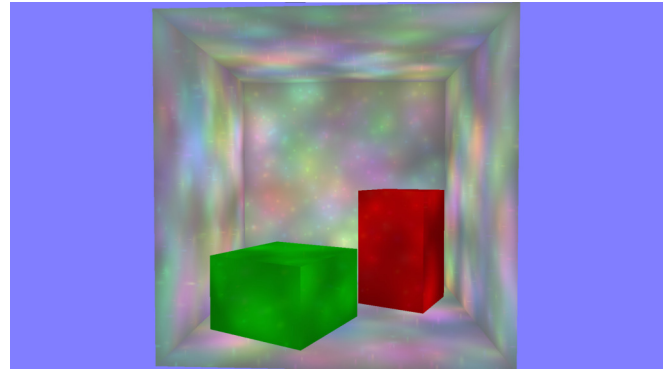


Fig. 6. *Cornell Box* test scene with 2000 lights spaced uniformly in the scene volume.

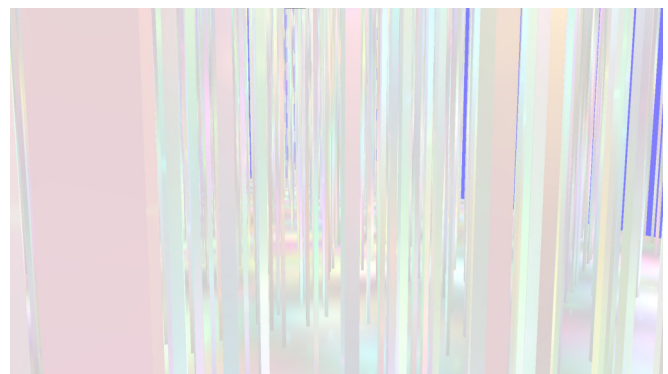


Fig. 7. *Bars* test scene with 50000 lights spaced uniformly in the scene volume.

with an Intel Core i7-9750H CPU 2.6 GHz and 16 GB RAM, supplied with an NVIDIA GeForce RTX 2060 Mobile (1920 CUDA cores) 6 GB GDDR6 with CC 7.5. All algorithms were implemented using C++17 with *Direct3D 11* [13] and *CUDA* [14] libraries and were implemented for the deferred shading pipeline.

The algorithms were tested on 4 different scenes with 3 different distributions of light positions. Images were rendered at a resolution of 1920×1080 pixels. In each test the camera moved along a predetermined path. To minimize the impact of the operating system's background work on rendering times, each test was repeated 5 times, and the results were averaged for each frame.

Three of the four test scenes, *Cornell Box*, *Sponza* [1] and *Rungholt* [1], are commonly used as a benchmark for evaluating rendering algorithms, and the other one, *Bars*, contains many vertical bars, and was created to test the algorithms on a scene containing many geometry discontinuities. All scenes, along with the light distribution used in them, are shown in Figs. 6, 7, 8 and 9.

The lights in all test scenes were distributed randomly, using one of three distributions: uniform distribution in the scene's volume; uniform distribution on the scene's geometry; groups of lights distributed uniformly in the scene's volume. The

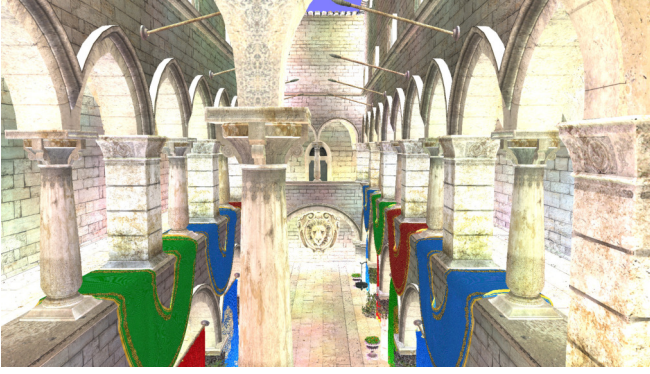


Fig. 8. *Sponza* [1] test scene with 50 000 lights spaced uniformly on the scene's geometry surface.

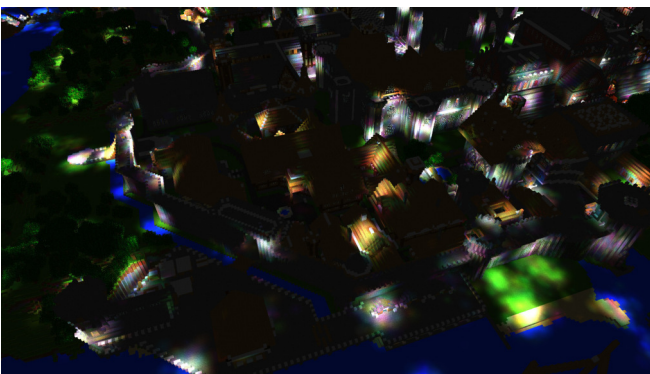


Fig. 9. *Rungholt* [1] test scene with 1 000 groups each containing 1 000 lights spaced uniformly in the scene volume.

lights' ranges were generated based on a uniform distribution between a minimum and maximum range, different for each scene. For the uniform distribution around the scene's volume, each coordinate of the light's position was generated independently of another, based on the scene's bounding box. In the uniform distribution around the scene's geometry, a random triangle from a fixed number (1 000 in the case of the performed tests) of the biggest triangles was selected at random, with the probability of being chosen proportional to the triangle's area. Then, a random point on a chosen triangle was generated and a light was placed in a position within the predefined distance of the chosen point along the triangle's normal vector. To generate groups of lights, the positions of the groups' centers were generated around the scene's volume. Then, for each group, a fixed number of lights' positions were generated using the truncated normal distribution [15] with the expected value equal to the generated group's center.

A number of tests were performed to determine the impact of each parameter on the average rendering time of each scene. In each test case, parameter configurations differed by a single parameter. As a base configuration we assumed:

- the division of the rendered image into 30×17 tiles;
- exponential depth division in the view-space;
- the unique cell list determination method using bitmasks;

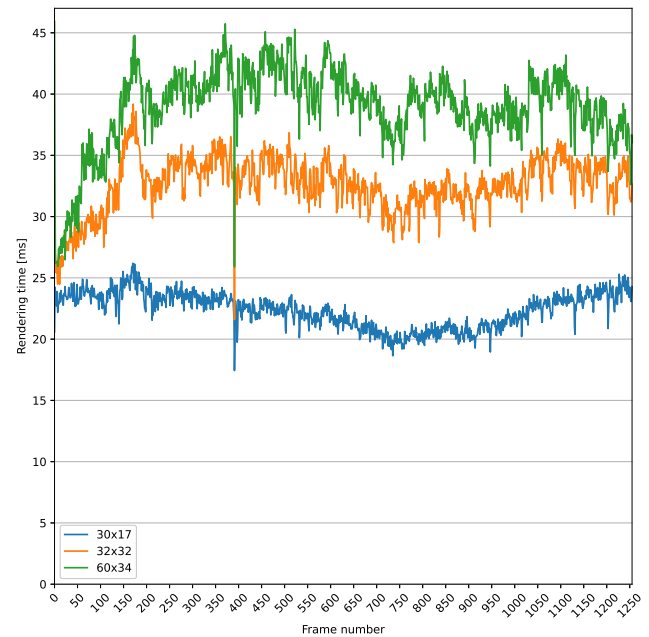


Fig. 10. Rendering time for each frame of the *Bars* test scene with different tile counts.

- precomputing the positions of cell corners.

Two different tile counts were compared with the base one: 32×32 and 60×34 tiles. In all test scenes, the lowest average time was achieved in the configuration using 30×17 tiles. The rendering times for each frame of the scene *Bars* are shown in Fig. 10. Table I shows the average results for all test scenes.

Two configurations with uniform depth division in the screen-space were tested: into 128 and into 256 cells. In the case of the *Cornell Box* and *Sponza* scenes, notably higher average rendering times were seen in the configuration with a higher cell number, whereas in the other two scenes a higher cell count resulted in lower average rendering times. Uniform depth division into 128 and 256 cells in the view-space was also tested. As with the screen-space division, whose cell count resulted in lower average rendering times, the rendering times differed between scenes. Notably, for the *Rungholt* scene the differences between the average rendering times were negligible. Table II shows the averaged results for all of the four described depth division methods for all test scenes.

Figs. 11, 12, 13 and 14 show the rendering times of each frame for the three tested methods of depth division. For the uniform divisions, a division resolution resulting in the lowest average rendering times was chosen. For the *Bars*, *Sponza* and *Rungholt* scenes, screen-space division resulted in significantly higher rendering times than the exponential division in the view-space. Uniform division in the view-space resulted in similar rendering times to exponential division for the *Cornell Box* and *Rungholt* scenes, but the rendering times for uniform division were higher than the exponential one for the *Bars* and *Sponza* scenes. In the case of the *Rungholt* scene

TABLE I
AVERAGE TIME [MS] FOR FRAME RENDERING USING THE DESCRIBED ALGORITHM FOR DIFFERENT IMAGE TILE COUNTS.

Test scene \ Tile count	30×17		32 × 32		60 × 34	
<i>Cornell Box</i>	11.07	15.61	+40.9%	18.71	+69.0%	
<i>Bars</i>	22.37	32.59	+45.7%	38.96	+74.1%	
<i>Sponza</i>	78.02	143.42	+83.8%	197.19	+152.8%	
<i>Rungholt</i>	63.32	84.10	+32.8%	108.58	+71.5%	

TABLE II
AVERAGE TIME [MS] OF FRAME RENDERING USING THE DESCRIBED ALGORITHM FOR FOUR DIFFERENT DEPTH DIVISIONS.

Test scene \ Cell count in depth	128 (screen-space)		256 (screen-space)		256 (view-space)	
<i>Cornell Box</i>	12.02	14.39	12.46	11.43		
<i>Bars</i>	40.97	39.78	20.03	26.23		
<i>Sponza</i>	99.44	106.25	60.14	73.66		
<i>Rungholt</i>	149.20	136.82	61.16	61.19		

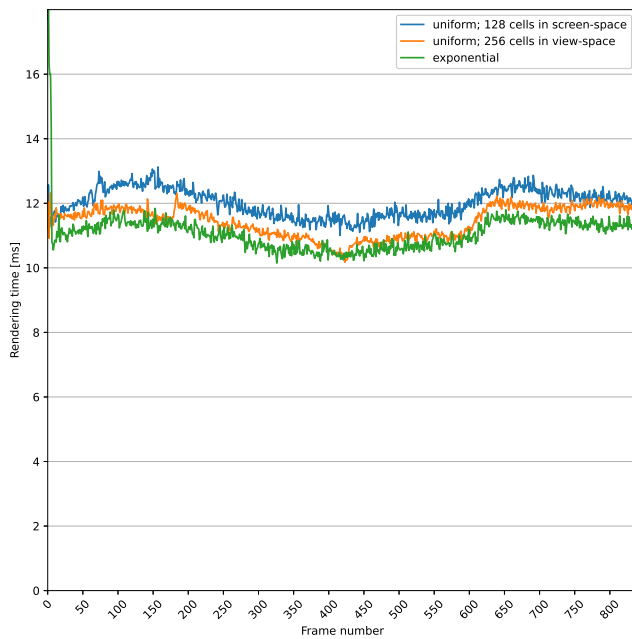


Fig. 11. Rendering time of each frame for the *Cornell Box* test scene for the three depth division methods.

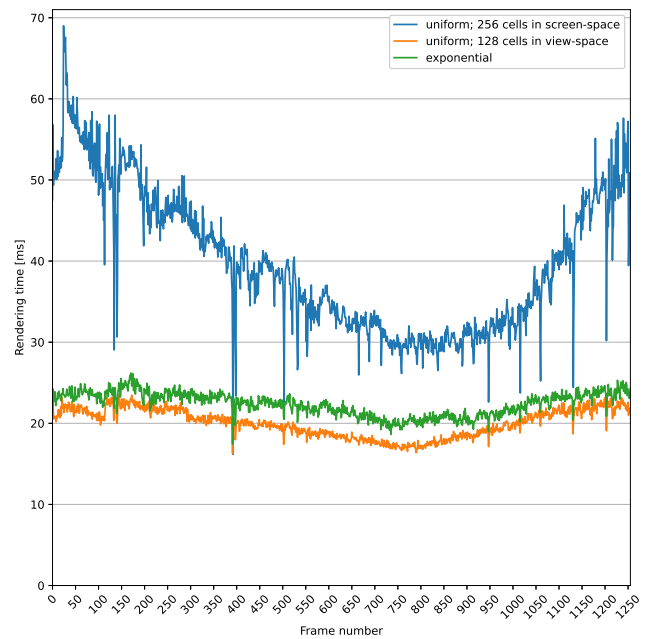


Fig. 12. Rendering time of each frame for the *Bars* test scene for the three depth division methods.

(Fig. 14), there were three parts of the test (at the beginning, in the middle and at the end), in which screen-space division resulted in significantly higher rendering times compared with the other methods. In these parts of the test, the camera was far from the scene geometry and a large area of the scene was visible.

Three methods of obtaining the list of cells filled with geometry were compared: using bitmaps, sorting and removing consecutive repeating elements, and removing consecutive repeating elements before sorting. These configurations were tested on the *Bars*, *Sponza* and *Rungholt* scenes. Fig. 15 shows the obtained results for the *Rungholt* scene. Table III shows

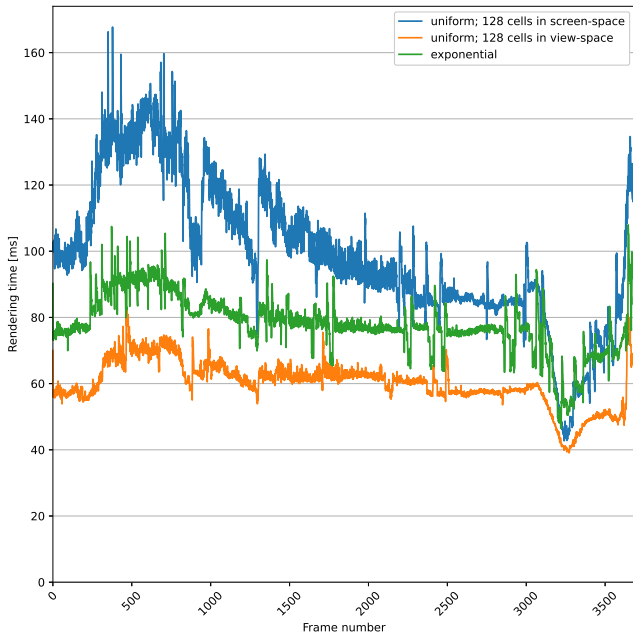


Fig. 13. Rendering time of each frame for the *Sponza* test scene for the three depth division methods.

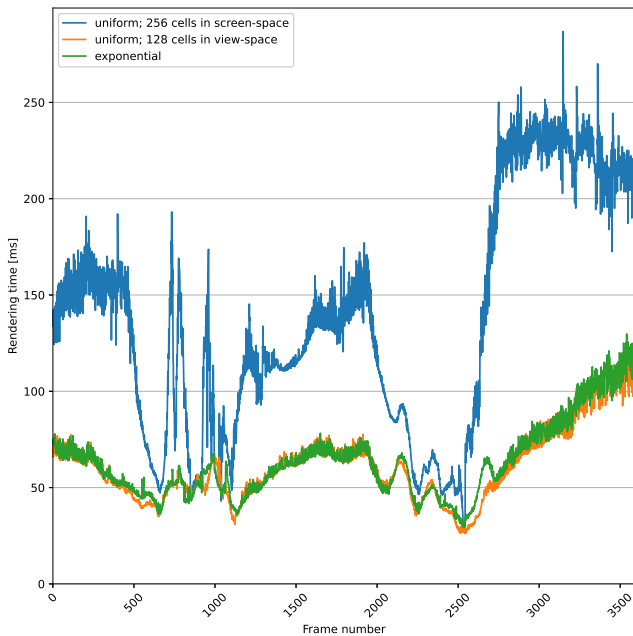


Fig. 14. Rendering time of each frame for the *Rungholt* test scene for the three depth division methods.

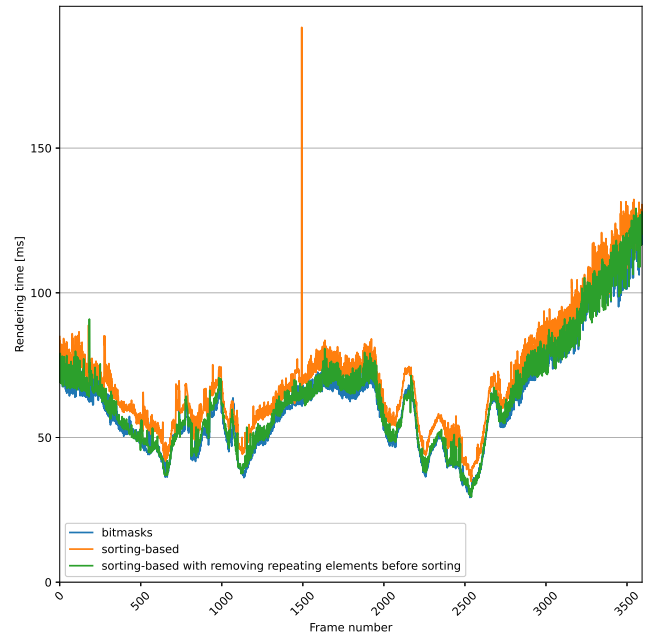


Fig. 15. Rendering time of each frame for the *Rungholt* test scene for the three methods of obtaining the list of unique cells.

the averaged results for all the test scenes. In all cases the lowest average time was achieved using the bitmasks and the highest time was seen using the sorting-based method without removing the repeating elements beforehand.

A version of the algorithm in which the cells’ corner positions were precomputed was also compared with a version in which the positions were calculated each time they were used. These configurations were also tested on the *Bars*, *Sponza* and *Rungholt* scenes. For all of these scenes, precomputing the cells’ corner positions resulted in significantly lower average rendering times than calculating them every time. Fig. 16 shows the results for the *Sponza* scene. Results for all of the tested scenes are shown in Table IV.

The proposed algorithm was compared in terms of rendering time and memory occupancy with the clustered shading and the original hybrid lighting algorithms. The algorithms were compared using the *Sponza* and *Rungholt* scenes with a variable number of lights, with lights being placed on the scene’s geometry in the *Sponza* scene, and with groups of lights in the *Rungholt* scene. Two configurations of the hybrid algorithm and the octree-based modification were tested, differing in the number of tiles used to divide the rendered image: 30×17 and 60×34 . This resulted in tiles of 32×32 and 64×64 pixels respectively. In the case of the clustered shading algorithm, tile sizes of 16×16 and 32×32 pixels were used. Depth was divided using the exponential division method, and a configuration of the clustered shading algorithm without using normal vectors during cluster creation was chosen.

Figs. 17 and 18 show the average rendering times for each tested algorithm variant. The octree-based algorithm

TABLE III

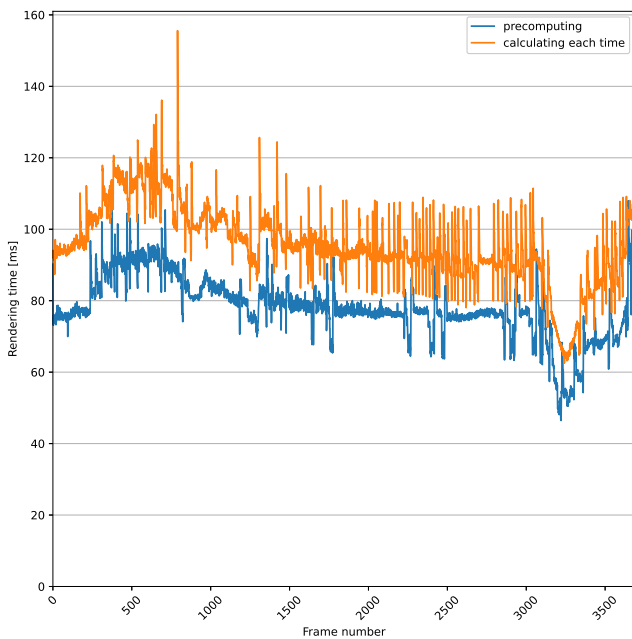
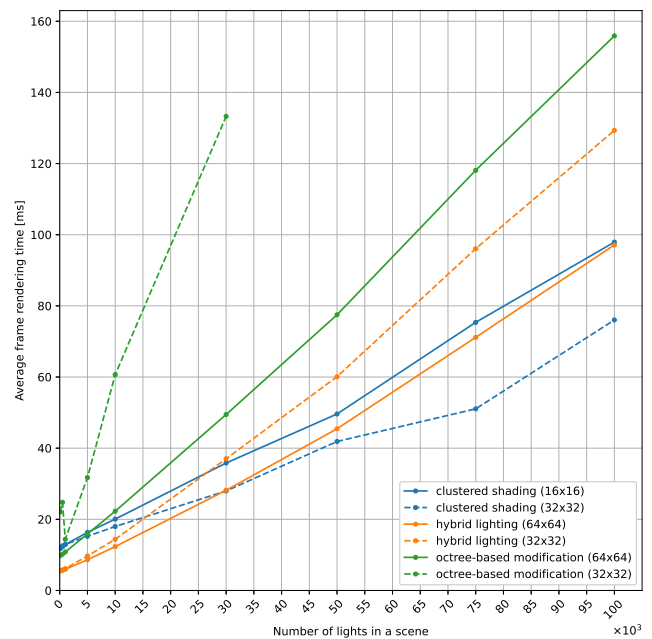
AVERAGE TIME [MS] OF FRAME RENDERING USING THE DESCRIBED ALGORITHM FOR THE THREE METHODS OF OBTAINING THE LIST OF UNIQUE CELLS.

Test scene	List obtaining method	bitmasks	sorting-based			
			sorting-based	sorting-based with the removal of repeating elements before sorting		
	<i>Bars</i>	22.37	29.13	+30.2%	24.07	+7.6%
	<i>Sponza</i>	78.02	84.29	+8.0%	78.78	+1.0%
	<i>Rungholt</i>	63.32	70.66	+11.6%	64.55	+1.9%

TABLE IV

AVERAGE TIME [MS] OF FRAME RENDERING USING THE DESCRIBED ALGORITHM FOR TWO METHODS USED TO CALCULATE CELLS' CORNER POSITIONS.

Test scene	Cells' corner position calculation method	precomputing	calculating each time	
			calculating each time	
	<i>Bars</i>	22.37	26.56	+18.8%
	<i>Sponza</i>	78.02	95.66	+22.6%
	<i>Rungholt</i>	63.32	69.05	+9.1%

Fig. 16. Rendering time of each frame for the *Sponza* test scene depending on the method used to calculate the cells' corner positions.Fig. 17. Average rendering time of the *Sponza* scene depending on the total number of lights in the scene, for each tested algorithm. The tile size, in pixels, is written in parentheses.

wasn't tested for the *Sponza* scene for the light count above 30 000 because of a rapidly rising rendering time for the increasing number of lights. In both test scenes the octree-based algorithm with a tile size of 32×32 saw significantly higher (by 11% – 392%) rendering times than the rest of the algorithms. The configuration with tiles of size 64×64 pixels, for up to 5 000 lights for the *Sponza* scene and up to 90 000 for the *Rungholt* scene, achieved lower (by 16% – 17%) rendering times than both configurations of the clustered shading algo-

rithm. For higher numbers of lights, the octree-based algorithm was slower than the clustered shading algorithm by up to 131% for the *Sponza* scene and up to 75% for the *Rungholt* scene.

Figs. 19 and 20 show the average sum of lights on the lists of lights depending on the total number of lights in a scene. As each light can be assigned to more than one list, the sum of light list elements may be bigger than the number of lights in a scene.

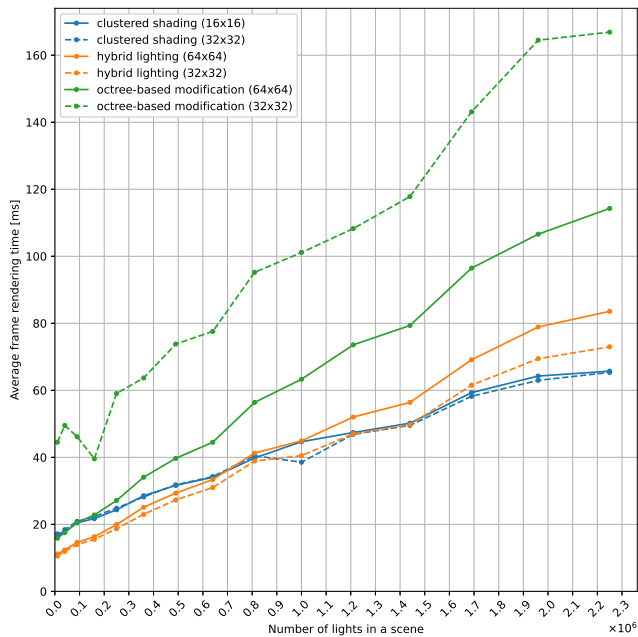


Fig. 18. Average rendering time of the *Rungholt* scene depending on the total number of lights in the scene, for each tested algorithm. The tile size, in pixels, is written in parentheses.

On average, the lowest total number of light list elements was achieved in the case of the octree-based algorithm with a tile size of 64×64 pixels. For the *Sponza* scene, there were at least 65% fewer elements compared with the other two algorithms, and for the *Rungholt* scene – at least 23% fewer. In the case of the *Sponza* scene, the proposed method with tiles measuring 32×32 pixels resulted in a lower total number of list elements (by 30% – 40%) than in the case of the original, unmodified version of the hybrid lighting algorithm with tiles that were twice as big.

V. SUMMARY

The results obtained in the tests show that both the scene geometry and the lights’ distribution are important factors impacting the rendering time.

Using an octree to store the lists of lights allows for a significant reduction (up to 65%) in the number of elements on the lists compared with the other tested algorithms. This resulted in fewer list insertion operations that needed to be performed for each frame. However, additional checks performed in order to determine whether or not a light could be stored in an internal node resulted in an overall slower algorithm than the original version.

The approach of using billboards to approximate the lights’ positions and ranges resulted in many calculations that were repeated by multiple threads. As each billboard’s pixel is potentially processed by a different thread, each thread has to check if the light affects all nodes in a group independently of other threads. One idea for modifying the described algorithm

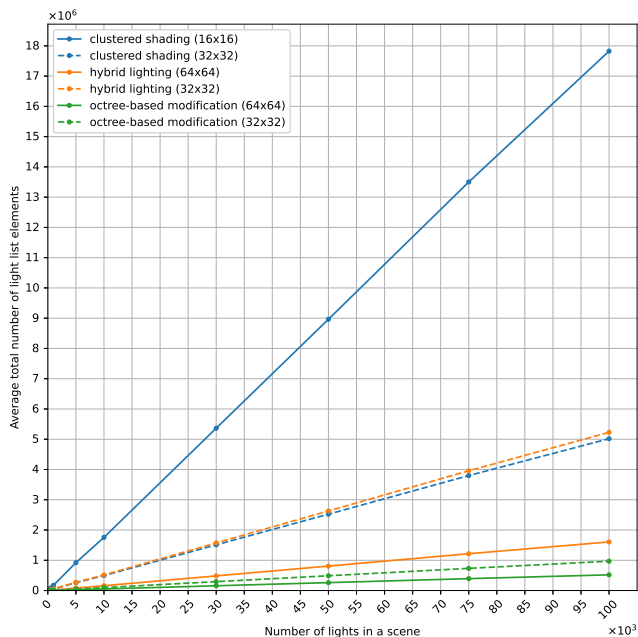


Fig. 19. Total number of light list elements averaged for all frames (*Sponza* scene) depending on the total number of lights in a scene, for each tested algorithm. The tile size, in pixels, is written in parentheses.

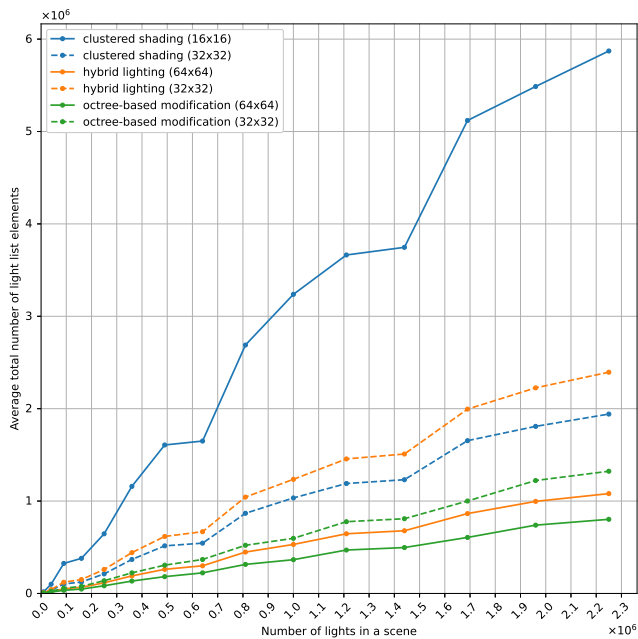


Fig. 20. Total number of light list elements averaged for all frames (*Rungholt* scene) depending on the total number of lights in a scene, for each tested algorithm. The tile size, in pixels, is written in parentheses.

is to adapt this method so that one light is processed entirely by a single thread.

Another modification that could result in shorter rendering times is to replace the sparse octree representation with full three-dimensional arrays, each representing one octree level. This modification would make it possible to read information about any node, without the necessity of tracing a path from the octree's root.

Another aspect of rendering scenes with many lights is accounting for multiple shadow sources. Shadow rendering poses a serious challenge, especially in the presence of many light sources, as both the rendering times [16] and shadow quality [17] have to be considered.

REFERENCES

- [1] M. McGuire, "Computer graphics archive," July 2017. [Online]. Available: <https://casual-effects.com/data>
- [2] A. Lauritzen, "Deferred rendering for current and future rendering pipelines," *SIGGRAPH Course: Beyond Programmable Shading*, pp. 1–34, 2010.
- [3] O. Olsson, E. Persson, and M. Billeter, "Real-time many-light management and shadows with clustered shading," in *ACM SIGGRAPH 2015 Courses*, 2015, pp. 1–398.
- [4] O. Olsson and U. Assarsson, "Tiled shading," *Journal of Graphics*, vol. GPU, pp. 235–251, 11 2011.
- [5] T. Harada, "A 2.5 d culling for forward+," in *SIGGRAPH Asia 2012 Technical Briefs*, 2012, pp. 1–4.
- [6] O. Olsson, M. Billeter, and U. Assarsson, "Clustered deferred and forward shading," in *Proceedings of the Fourth ACM SIGGRAPH/Eurographics conference on High-Performance Graphics*. Citeseer, 2012, pp. 87–96.
- [7] J. Archer, G. Leach, P. Knowles, and R. van Schyndel, "Hybrid lighting for faster rendering of scenes with many lights," *The Visual Computer*, vol. 34, no. 6, pp. 853–862, 2018.
- [8] J. Dupuy, J.-C. Iehl, and P. Poulin, *Quadrees on the GPU*, 10 2018, pp. 211–222.
- [9] D. Wehr and R. Radkowski, "Parallel kd-tree construction on the gpu with an adaptive split and sort strategy," *International Journal of Parallel Programming*, vol. 46, no. 6, pp. 1139–1156, 2018.
- [10] J. R. Jørgensen, K. Scheel, and I. Assent, "Gpu-inscy: A gpu-parallel algorithm and tree structure for efficient density-based subspace clustering," in *EDBT*, 2021, pp. 25–36.
- [11] C. Crassin, F. Neyret, M. Sainz, S. Green, and E. Eisemann, "Interactive indirect illumination using voxel cone tracing," in *Computer Graphics Forum*, vol. 30, no. 7. Wiley Online Library, 2011, pp. 1921–1930.
- [12] Y. O'Donnell and M. G. Chajdas, "Tiled light trees," in *Proceedings of the 21st ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, 2017, pp. 1–7.
- [13] Microsoft, "Direct3d 11 website," 2022. [Online]. Available: <https://docs.microsoft.com/en-us/windows/win32/direct3d11/atoc-dx-graphics-direct3d-11>
- [14] NVIDIA Corporation, "Cuda toolkit website," 2022. [Online]. Available: <https://developer.nvidia.com/cuda-toolkit>
- [15] J. Burkhart, "The truncated normal distribution," *Department of Scientific Computing Website, Florida State University*, pp. 1–35, 2014.
- [16] O. Olsson, E. Sintorn, V. Kämpe, M. Billeter, and U. Assarsson, "Efficient virtual shadow maps for many lights," in *Proceedings of the 18th Meeting of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, ser. I3D '14. New York, NY, USA: Association for Computing Machinery, 2014, p. 87–96. [Online]. Available: <https://doi.org/10.1145/2556700.2556701>
- [17] K. Kluczek, "Quality metric for shadow rendering," in *2016 Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2016, pp. 791–796.

Encrypting JPEG-compressed Images by Substituting Huffman Code Words

Marek Parfieniuk

University of Bialystok, Institute of Computer Science
ul. Konstantego Ciolkowskiego 1M, 15-245 Bialystok, Poland
Email: marek.parfieniuk@uwb.edu.pl

Abstract—This paper presents a method for encrypting JPEG-coded images that preserves both compression ratio and format of a bit stream. Such solutions allow for selectively hiding information: image contents can be encrypted, while in-file meta-data remain readable. Our algorithm is a symmetric, polygram substitution cipher, as it replaces Huffman code words and rearranges value bits that describe the main results of the Discrete Cosine Transform (DCT) of a pixel block: the DC coefficient and the first non-zero AC coefficient. Both length and format of a file are preserved, because bits are modified under constraints on their numbers. Such encryption is a kind of post-processing of a compressed bit stream, and thus it can be built on the top of an existing JPEG codec, without accessing its internals. Compared to previous similar solutions, our approach better hides image contours, exchanging AC for DC energy. Our work also reveals some properties of Huffman code tables and bit streams related to the JPEG standard.

I. INTRODUCTION

IN RECENT years, one can notice research efforts to combine the JPEG standard for image coding with data encryption [1]. Format-preserving approaches to joint encryption-compression are less efficient, more difficult to implement, and less secure than general-purpose ciphers. Nevertheless, they are useful, allowing for selective encryption, i.e. for protecting only a subset of information contained in a file [2].

In this paper, we propose a method for format-compliant encryption of JPEG files that is based on substituting and restructuring pairs of variable-length code words under constraints on the total number of bits in a pair. These modifications are made to only code words related to the main results of the Discrete Cosine Transform (DCT) and quantization: to the DC coefficient and to the first non-zero AC coefficient.

As a couple of data units is replaced in accordance with a cryptographic key, our solution is a polygram substitution cipher. Consisting in post-processing of encoding results, it can be built on the top of an existing JPEG codec, without accessing its internals.

Similar known approaches, [3]–[5], modify DC coefficients separately from AC ones. So they have little effects on image contours, unless coefficients are exchanged among blocks of 8×8 pixels. Our solution is able to exchange DC for AC energy of one image block, so as to better hide the latter, buffering the minimum number of bits.

This paper additionally reveals some new facts on the default Huffman dictionaries and on statistics of code words in JPEG bit streams.

II. ENTROPY CODING OF DCT COEFFICIENTS IN THE JPEG STANDARD

The JPEG standard specifies that an image to be compressed is divided into blocks of 8×8 pixels, and each block is converted into a vector of 64 coefficients, which then are entropy coded. The conversion consists in computing the 2-D DCT of the block, quantizing the resulting matrix, and taking quantized elements in the zig-zag order.

The first of 8×8 DCT outputs is called the DC (direct-current) coefficient, as it is the average value of all pixels of a block. The remaining 63 outputs are called AC (alternate-current) coefficients, because they reflect deviations from the average value, of various frequencies.

For two consecutive blocks of 8×8 pixels, the average pixel intensities, or the DC coefficients, often have similar values. Therefore, it is advantageous to encode the difference between them, by using two variable-length code words. The first one has a length of $2 \leq h_0 \leq 9$ bits and results from Huffman encoding of $0 \leq v_0 \leq 11$, the index of the value category that embraces the difference value. The second code word comprises v_0 bits which point out a particular value in the category. The first bit represents the sign of the difference, whereas the remaining bits determine its magnitude.

Table I lists the categories and Huffman code words used to encode DC coefficients. The lower magnitudes of the values that comprise a category, the fewer values in this category. So, a shorter the code word is assigned to its index, in order to achieve data compression.

A non-zero quantized AC coefficient is usually preceded by a series of zero-valued ones. Thus its value is encoded together with the number of the latter, by using two code words. For the k th non-zero coefficient, in the zig-zag order, the first word comprises $2 \leq h_k \leq 16$ bits and is obtained by Huffman encoding of the (r_k, v_k) pair, where $0 \leq r_k \leq 15$ is the number of the zero-valued AC coefficients that precede this non-zero one, and $1 \leq v_k \leq 10$ is the index of the category that embraces the value of this coefficient. The second word comprises v_k bits, which determine a value in the v_k th category. As zero-valued quantized AC coefficients often occur in long runs, and non-zero ones often have small magnitudes, compression can be achieved by assigning shorter words to (r_k, v_k) pairs that describe such occurrences.

Two special Huffman code words occur without value bits.

TABLE I
VALUE CATEGORIES AND HUFFMAN CODE WORDS FOR ENCODING DIFFERENCES BETWEEN DC COEFFICIENTS OF IMAGE LUMINANCE

v_0	Value range	Huffman code word	h_0
0	0	00	2
1	± 1	010	3
2	-3, -2, 2, 3	011	3
3	$\pm(4, \dots, 7)$	100	3
4	$\pm(8, \dots, 15)$	101	3
5	$\pm(16, \dots, 31)$	110	3
6	$\pm(32, \dots, 63)$	1110	4
7	$\pm(64, \dots, 127)$	11110	5
...
11	$\pm(1024, \dots, 2047)$	11111110	9

The EOB (End-of-Block) word is placed after the codes of the last non-zero AC coefficient, so as to tell that the remaining ones are zero. The ZRL (Zero-run-length) word represents a series of 16 zero AC coefficients between non-zero ones.

III. ENCRYPTING JPEG BIT STREAMS BY SUBSTITUTING HUFFMAN CODE WORDS

A. Substitution idea and constraints

Polygram substitution ciphers replace several symbols of a text with the same number of other letters. Following this idea, we have shown in [6] that Huffman-encoded data can be encrypted by substituting pairs of code words. Herein, we adapt this approach to JPEG bit streams, in which Huffman code words are interleaved with value bits, and thus they need to be substituted somewhat tricky. Moreover, it is pointless to modify all code words, as the contents of an image is described primarily by the main quantized coefficients of the DCT of an 8×8 pixel block: the DC one and the first non-zero AC one.

So, we propose to encrypt JPEG-encoded images by processing them block-by-block, by substituting Huffman code words that describe categories of the aforementioned coefficients and by moving bits between the code words that describe coefficient values. As the cipher should not enlarge files, we allow only such substitutions that

$$\underline{h}_0 + \underline{h}_1 = h_0 + h_1 \quad \text{and} \quad \underline{v}_0 + \underline{v}_1 = v_0 + v_1 \quad (1)$$

where underlines denote the lengths of code words that replace the original ones. These constraints ensure that a substitution changes neither the total length of Huffman code words nor the total number of value bits, $\underline{h}_0 + \underline{h}_1 + \underline{v}_0 + \underline{v}_1 = v_0 + v_1 + h_0 + h_1$.

The constraints can be explained by using Table II. It lists 12-bit, equal-length combinations of DC- and AC-related code words and value bits. The combinations have been grouped with respect to the total number of the value bits, $v_0 + v_1$. A combination can be substituted for each other of the same group, but not for one of another group.

B. Encryption procedure

Both encryption and decryption of our cipher can be explained by using the data flow shown in Fig. 1.

Assuming that the method is applied to an existing JPEG bit stream, the first step is to scan the stream in order to determine the Huffman code words and value bits that describe the DC and AC coefficients of interest of a subsequent block of 8×8

TABLE II
ALL 12-BIT COMBINATIONS OF DC- AND (AFTER "-") AC-RELATED HUFFMAN CODE WORDS AND VALUE BITS (DENOTED AS "x", BEING 0 OR 1), GROUPED WITH RESPECT TO THE TOTAL NUMBER OF VALUE BITS

DC-AC code words	h_0	h_0	v_0	v_1	r_1	$v_0 + v_1$	$h_0 + h_1$
00-111111000x	2	9	0	1	8	1	11
00-111111001x	2	9	0	1	9		
00-111111010x	2	9	0	1	10		
00-11111001xx	2	8	0	2	2	2	10
010x-1111010x	3	7	1	1	5		
010x-1111011x	3	7	1	1	6		
00-1111001xxx	2	7	0	3	1		
011xx-111010x	3	6	2	1	3	3	9
011xx-111011x	3	6	2	1	4		
011xx-11011xx	3	5	2	2	1	4	8
100xxx-11100x	3	5	3	1	2		
00-11010xxxxx	2	5	0	5	0		
011xx-1011xxx	3	4	2	3	0	5	7
101xxxx-1100x	3	4	4	1	1		

pixels. If the method has to be integrated with a JPEG encoder, than it can be applied to quantized coefficients before forming a bit stream. Knowing the code words, or coefficient values, one can determine $(h_1 + h_0)$ and $(v_1 + v_0)$ and decide whether substitution is possible. If so, then these bit counts point out the group of $N \geq 2$ pairs of code words that can be exchanged for each other in accordance with (1).

Let us assume that the members of the substitute group are ordered, so that they form an array and can be indexed by the numbers from 0 to $(N - 1)$. If i_{original} is the index of the pair of original code words, than by adding (modulo N) a pseudorandom offset we obtain the index of a substitute pair:

$$i_{\text{substitute}} = \text{mod}(i_{\text{original}} + \text{randi}(N), N) \quad (2)$$

The "mod" function gives the remainder after division, and is used to ensure that $0 \leq i_{\text{substitute}} < N$, like i_{original} . The "randi" produces a pseudorandom integer from the discrete uniform distribution over $[1, N]$. This function is assumed to use a pseudorandom number generator whose output can be controlled by using the cryptographic key as the seed.

The Huffman code words determined by $i_{\text{substitute}}$ are submitted to the output (encrypted) bit stream, followed by the original value bits, rearranged in accordance with \underline{v}_0 and \underline{v}_1 that match the substitutes. The cipher does not affect bits that describe the rest of AC coefficients of the given pixel block.

C. Decryption procedure

By detecting and analysing a pair of DC- and AC-related code words of the encrypted bit stream, one can determine the substitute group and $i_{\text{substitute}}$. The original code words are pointed by the inverse of (2)

$$i_{\text{original}} = \text{mod}(i_{\text{substitute}} - \text{randi}(N), N) \quad (3)$$

provided that, at both encryption and decryption sides of a flow of JPEG-compressed pictures, (i) members of a substitute group are ordered in the same way, (ii) the same cryptographic key is used as the seed of the pseudorandom generator behind the "randi" function, when the processing of a bit stream starts, and (iii) 8×8 pixel blocks are processed in the same order.

Obviously, in addition to recovering the Huffman code words, it is necessary to accordingly rearrange the value bits.

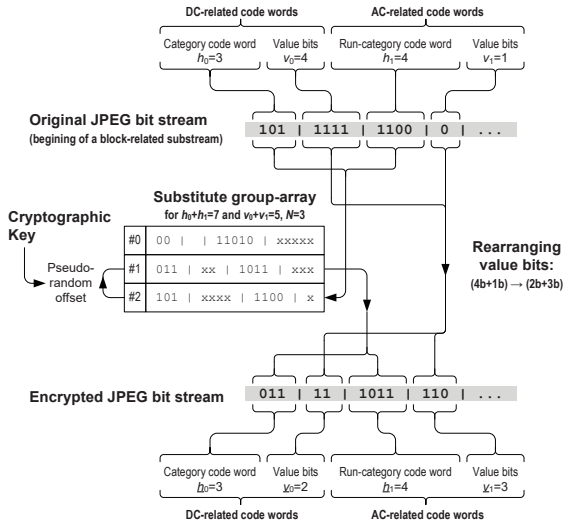


Fig. 1. Encryption of JPEG bit streams by length-constrained substitution of Huffman code words and rearrangement of value bits.

IV. SUBSTITUTION LIMITATIONS

For most images and quality settings, code words can be substituted sufficiently often to make it impossible to recover a readable, quality image from an encrypted bit stream, without knowing the cryptographic key. However, when an image is encoded with low-quality settings, pixel blocks might occur to which the proposed cipher cannot be applied. The default Huffman dictionaries determine not so many pairs of code words that have no substitutes, but the majority of the replaceable combinations of bits occur only occasionally when compressing natural images.

A. Substitution limitations by Huffman code tables

The JPEG standard specifies the default dictionaries of Huffman code words. The tables for luminance contain 12 words for encoding differences between DC coefficients and 161 words for encoding AC coefficients. These words can be combined into $12 \times 161 = 1932$ DC-AC pairs, which can be grouped with respect to both the total length of Huffman code words and the total number of the value bits that must accompany them. If a resulting group comprises two or more pairs of code words, then these DC-AC pairs can be substituted for each other in accordance with (1).

The sizes of the groups can be visualized by colors as in Fig. 2. The bit numbers related to the axes determine a group, and are coordinates of the small square whose color reflects the number of pairs in this group, in accordance with the legend.

Groups exist such that a pair of code words has as many as several dozen of substitutes. But some groups comprise only one combination of Huffman code words, which cannot be substituted. The great majority of the pairs, 1892 (98%) of them, have at least one potential substitute.

Table III lists the DC-AC pairs of code words that cannot be substituted. Moreover, the cipher cannot be applied to blocks for which all quantized AC coefficients are zero, i.e. when the DC-related bits are followed by the EOB special code word.

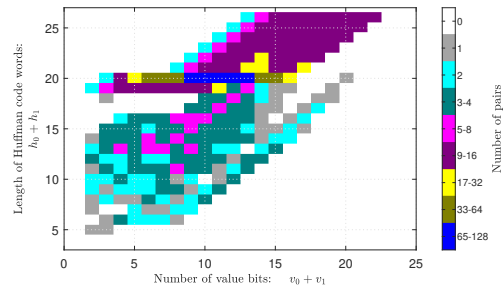


Fig. 2. Numbers of DC-AC pairs of default code words that can be substituted in accordance with (1).

TABLE III
PAIRS OF DC- AND (AFTER "-") AC-RELATED DEFAULT CODE WORDS THAT CANNOT BE SUBSTITUTED IN ACCORDANCE WITH (1)

#	DC- and AC-related code words and value bits (denoted as "x")	h_0	v_0	h_1	v_1	r_1	$h_0 + h_1$	$v_0 + v_1$	$h_0 + v_0 + h_1 + v_1$
1	00-00x	2	0	2	1	0	4	1	5
2	00-01xx	2	0	2	2	0	4	2	6
3	010x-00x	3	1	1	2	0	5	2	7
4	011xx-1100x	3	2	4	1	1	7	3	10
5	100xxx-1100x	3	3	4	1	1	7	4	11
6	011xx-100xxx	3	2	3	3	0	6	5	11
7	100xxx-100xxx	3	3	3	3	0	6	6	12
8	110xxxx-01xx	3	5	2	2	0	5	7	12
9	100xxx-1011xxxx	3	3	4	4	0	7	7	14
10	100xxx-11010xxxx	3	3	5	5	0	8	8	16
11	1111110xxxxxxx-00x	7	9	2	1	0	9	10	19
12	101xxxx-1111000xxxxxx	3	4	7	6	0	10	10	20

B. Substitution limitations by image contents

The essence of the JPEG standard lays in the adjustment of entropy codes to statistics of quantized DCT coefficients of an average, natural image. By lowering the quality settings, one quantizes DCT coefficients more roughly, and thus decreases differences between DC coefficients, increases the number and seriality of zero-valued AC coefficients, and decreases magnitudes of non-zero ones. The reconstructed image looks worse, but it is described by a shorter bit stream.

Shortening of code words results in an increased probability that they cannot be replaced in accordance with our cipher. Moreover, it decreases the number of substitutes, when code words can be replaced. Both these issues can be explained by using Table IV and Fig. 3, which show properties of bit streams that result from JPEG-encoding of the Lena image for various quality settings.

In Table IV, one can see that decreasing quality increases the number of pixel blocks which are described by only DC-related bits, followed by the EOB code word, so that our cipher cannot be applied. Such situations do not occur for higher-quality settings, $Q > 75\%$, but become frequent for $Q < 50\%$.

In Fig. 3, the color of a small square shows how many blocks of an image are described by code words and value bits, whose lengths and numbers, respectively, are given by the coordinates of the square. When the quality is decreased, the distribution of code words shifts toward the origin: shorter ones occur more frequently, while longer ones disappear.

The problem becomes clear after comparing the plots in Fig. 3 to that in Fig. 2. The former overlap little with the

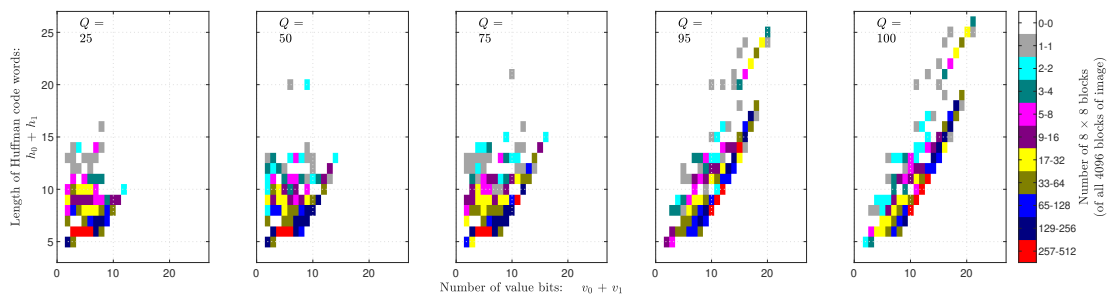


Fig. 3. Occurrence frequencies of DC-AC pairs of code words for the Lena image (512×512 pixels).

TABLE IV
PROPERTIES OF JPEG BIT STREAMS OF THE LENA IMAGE

Quality [%]	Ratio [bpp]	PSNR [dB]	Percentage of blocks with the DC coefficient followed by N nonzero AC coefficients [%]				
			$N = 0$	$N = 1$	$N = 2$	$N = 3$	$N \geq 4$
100	4.93	47.34	0	0	0	0	100
90	1.78	39.92	0	0	0	0.02	99.98
75	0.98	37.40	0.29	1.46	4.27	6.98	86.98
50	0.63	35.55	6.22	11.49	13.62	10.88	57.76
25	0.43	33.57	22.31	19.11	13.28	8.10	37.18

TABLE V
PERCENTAGES OF 8×8 PIXEL BLOCKS FOR WHICH THE DC-AC CODE WORDS CAN BE SUBSTITUTED IN ACCORDANCE WITH OUR CIPHER

Quality [%]	Lena	Baboon	Barbara	Boat
	Percentage of blocks [%]			
100	84.49	88.84	85.81	84.17
90	84.47	88.33	85.37	78.83
75	81.98	80.85	80.46	71.02
50	69.60	73.04	69.82	55.93
25	51.63	68.60	57.39	43.77

latter even for higher-quality compression. For lower-quality settings, the overlap is even smaller, and, what is even worse, it occurs at the region, in Fig. 2, that is related to code words that cannot be substituted or belong to substitute groups that comprise very few pairs of code words.

So, in order to evaluate the probability that our cipher can be applied to a pixel block of a given image, for given quality settings, one should determine the corresponding data distributions like those illustrated in Fig. 3 and combine it with that that is shown in Fig. 2.

Table V shows summaries of such evaluations for several well-known test images. In particular, it lists the percentages of 8×8 blocks to which our cipher can be applied. They are satisfactory: for reasonable quality settings, our cipher is able to modify considerable areas of an image.

A related measure of security is the average number of substitutes per block. For these images, it varies from 1.3–1.9 to 2.9–3.0, for quality settings from 25% to 100%, respectively. By raising these numbers to the power of the number of blocks in an image, one can estimate the number of substitution combinations that must be reviewed in a brute force attack on our cipher. Obviously, it is virtually impossible to recover a quality image, without knowing the secret key.

V. CIPHER EVALUATION

A. Conceptual contribution

Our idea is related to encrypting data by altering Huffman tables in accordance with a cryptographic key [7], [8]. However, we noticed no similar solutions, which would be based on modifying both Huffman code words and value bits, of DC and AC DCT coefficients. In most works, AC- and DC-related code words are transformed independently, so encryption-related changes to the bit stream are less deep than in our cipher, unless coefficients are exchanged among blocks of 8×8 pixels, which requires complicated data buffering.

By modifying together DC and AC coefficients of one pixel block, our method is able to more affect the latter, or image contours, without referring to other blocks.

A value of our works is also in providing arguments to question the claims of [9], in which Huffman coding has been evaluated as being unsuitable to encryption aimed at format compliance and file size preservation.

B. Cipher manifestation in images and its strength

Figure 4 shows the Lena test image, and results of straightforward (without decryption) decoding of a corresponding JPEG bit stream that had been encrypted using our method. The reconstructed image is unreadable, mainly because of rapid changes in average intensity of pixel blocks.

Unfortunately, an attacker can easily retrieve contours of images from encrypted files. It is sufficient to only set all DC coefficients to zero, and then to decode a picture without care about decryption. Fig. 5 shows results of such decoding of the Lena image from original and encrypted bit streams. Even though edges are reconstructed incorrectly, they appear in correct blocks, forming contours.

This weakness is not specific for our cipher. It characterizes virtually all JPEG-compliant and file-size-preserving approaches to joint compression-encryption, being related to the JPEG coding principle of processing images block-by-block. A cipher cannot modify the contour location without moving information from block to block. So most publications about extending JPEG coding with encryption describe some method of exchanging pixels or (encoded) DCT coefficients among blocks [3], [5]. They destroy contours but at the costs of buffering an entire image and of increasing file size. Our solution could as well be combined with such a method.



Fig. 4. Lena image and corresponding picture decoded without taking into consideration that JPEG bit streams have been encrypted.

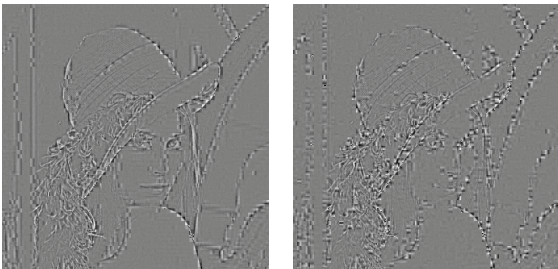


Fig. 5. Non-DC contents of the Lena images: original and decoded without taking into consideration that JPEG bit streams have been encrypted.

C. Inconsistencies in DC and AC coefficients

Our cipher might produce files that are logically inconsistent. In Fig. 4, artifacts result mainly from that encrypted code words might describe differences that sum up to values of the DC coefficient that are outside the allowed range $\pm 2^{11}$, for luminance. A decoder silently converts an incorrect value into a number in the range, by overflow, or saturation.

The issue is known, and some methods have been proposed to handle it [3], [10], but they are based on simultaneously processing many blocks of 8×8 pixels. The problem seems to be neglected in some works, in which DC-related Huffman code words are left untouched, and only value bits are modified simply, by pseudorandomly changing only value sign or by XOR-ing more bits with pseudorandom patterns [11], [12].

In most of works that modify the DC coefficient, the file size is not preserved [13]–[15].

A related problem is that encrypted AC code words might describe so long runs of zero-valued AC coefficients, that the 64-element vector is too short to put all coefficients into it. We noticed no publications which would discuss this, even though the issue can be caused by some known ciphers, like those of [4] and [16] that shuffle AC-related code words inside a block and among blocks, respectively.

D. Computational load and memory consumption

Considerable computations or memory are necessary to determine substitutes of code words. A slower but memory-efficient approach is to determine a substitute on-the-fly, by scanning Huffman tables provided by a JPEG codec. Alternatively, an array of arrays can be prepared that stores information about substitute groups. It would occupy an additional

memory much larger than that of the Huffman dictionary, but substitutions could be realized quickly by using $(h_1 + h_0)$ or $(v_1 + v_0)$ as an index into the array of code-word groups.

VI. CONCLUSION

From the point of view of practice, our cipher rather only slightly outperforms the known approaches to combining JPEG compression with encryption. However, it can be evaluated as conceptually subtle and sophisticated, being based on nuances related to JPEG Huffman dictionaries of code words and to distributions of quantized DCT coefficients. To the best of our knowledge, this is the first paper that demonstrates and analyses these nuances, via unique plots and tables.

REFERENCES

- [1] P. Li and K.-T. Lo, "Survey on JPEG compatible joint image compression and encryption algorithms," *IET Signal Process.*, vol. 14, no. 8, pp. 475–488, 2020. doi: 10.1049/iet-spr.2019.0276
- [2] A. Massoudi, F. Lefebvre, C. De Vleeschouwer, B. Macq, and J.-J. Quisquater, "Overview on selective encryption of image and video: Challenges and perspectives," *EURASIP J. Inf. Secur.*, vol. 2008, pp. 5:1–5:18, Jan. 2008. doi: 10.1155/2008/179290
- [3] J. He, S. Huang, S. Tang, and J. Huang, "JPEG image encryption with improved format compatibility and file size preservation," *IEEE Trans. Multimedia*, vol. 20, no. 10, pp. 2645–2658, 2018. doi: 10.1109/TMM.2018.2817065
- [4] S. Auer, A. Bliem, D. Engel, A. Uhl, and A. Unterwiesing, "Bitstream-based JPEG encryption in real-time," *Int. J. Digital Crime Forensics*, vol. 5, no. 3, pp. 1–14, Jul. 2013. doi: 10.4018/jdcf.2013070101
- [5] K. Kurihara, M. Kikuchi, S. Imaizumi, S. Shiota†, and H. Kiya, "An encryption-then-compression system for JPEG/Motion JPEG standard," *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, vol. E98.A, no. 11, pp. 2238–2245, Nov. 2015. doi: 10.1587/transfun.E98.A.2238
- [6] M. Parfieniuk and P. Jankowski, "Encrypting Huffman-encoded data by substituting pairs of code words without changing the bit count of a pair," in *Proc. 3rd Int. Conf. Cryptography Security Syst. (CSS)*, Lublin, Poland, 22–24 Sep. 2014. doi: 10.1007/978-3-662-44893-9_2 pp. 12–22.
- [7] M. S. Kankanhalli and T. T. Guan, "Compressed-domain scrambler/descrambler for digital video," *IEEE Trans. Consum. Electron.*, vol. 48, no. 2, pp. 356–365, May 2002. doi: 10.1109/TCE.2002.1010142
- [8] C.-P. Wu and C.-C. J. Kuo, "Fast encryption methods for audiovisual data confidentiality," in *Multimedia Syst. Appl. III*, ser. Proc. SPIE, vol. 4209, Boston, MA, Nov. 2000. doi: 10.1117/12.420829 pp. 284–295.
- [9] S. Li, *On the Performance of Secret Entropy Coding: A Perspective Beyond Security*. Berlin, Heidelberg: Springer, 2012, pp. 389–401.
- [10] W. Li and Y. Yuan, "A leak and its remedy in JPEG image encryption," *Int. J. Computer Mathematics*, vol. 84, no. 9, pp. 1367–1378, Sep. 2007. doi: 10.1080/00207160701294376
- [11] K. Yi and K. Kim, "Encryption method of compressed images with JPEG compliance by shuffling information both in spatial and frequency domains," in *Advanced Multimedia and Ubiquitous Engineering*, J. J. Park, H. Jin, Y.-S. Jeong, and M. K. Khan, Eds. Singapore: Springer, 2016. doi: 10.1007/978-981-10-1536-6_86 pp. 661–667.
- [12] S. Li and Y. Zhang, "Quantized DCT coefficient category address encryption for JPEG image," *KSII Trans. Internet Inf. Syst.*, vol. 10, no. 4, pp. 1790–1806, Apr. 2016. doi: 10.3837/tis.2016.04.018
- [13] Y. Mao and M. Wu, "A joint signal processing and cryptographic approach to multimedia encryption," *IEEE Trans. Image Process.*, vol. 15, no. 7, pp. 2061–2075, Jul. 2006. doi: 10.1109/TIP.2006.873426
- [14] S. Lian, J. Sun, and Z. Wang, "A novel image encryption scheme based on JPEG encoding," in *Proc. 8th Int. Conf. Inf. Vis. (IV)*, London, UK, 16 Jul. 2004. doi: 10.1109/IV.2004.1320147 pp. 217–220.
- [15] X. Niu, C. Zhou, J. Ding, and B. Yang, "JPEG encryption with file size preservation," in *Proc. Int. Conf. Intell. Inf. Hiding Multimedia Signal Process. (IIHMSP)*, Harbin, China, 15–17 Aug. 2008. doi: 10.1109/IIHMSP.2008.207 pp. 308–311.
- [16] B. Yang, C.-Q. Zhou, C. Busch, and X.-M. Niu, "Transparent and perceptually enhanced JPEG image encryption," in *Proc. 16th Int. Conf. Digital Signal Process.*, Santorini, Greece, 5–7 Jul. 2009. doi: 10.1109/ICDSP.2009.5201075 pp. 1–6.

12th Workshop on Scalable Computing

THE world of large-scale computing continuously evolves. The most recent addition to the mix comes from numerous data streams that materialize from exploding number of cheap sensors installed “everywhere”, on the one hand, and ability to capture and study events with systematically increasing granularity, on the other. To address the needs for scaling computational and storage infrastructures, concepts like: edge, fog and dew computing emerged.

Novel issues involved in “pushing computing away from the center” did not replace open questions that existed in the context of grid and cloud computing. Rather, they added new dimensions of complexity and resulted in the need of addressing scalability across more and more complex ecosystems consisting of individual sensors and micro-computers (e.g. Raspberry PI based systems) as well as supercomputers available within the Cloud (e.g. Cray computers facilitated within the MS Azure Cloud).

Addressing research questions that arise in individual “parts” as well as across the ecosystem viewed from a holistic perspective, with scalability as the main focus is the goal of the Workshop on Scalable Computing. In this context, the following topics are of special interest (however, this list is not exhaustive).

TOPICS

- General issues in scalable computing
 - Algorithms and programming models for large-scale applications, simulations and systems
 - Large-scale symbolic, numeric, data-intensive, graph-oriented, distributed computations
 - Fault-tolerant and consensus techniques for large-scale computing
 - Resilient large-scale computing
 - Data models for large-scale applications, simulations and systems
 - Large-scale distributed databases
 - Load-balancing / intelligent resource management in large-scale applications, simulations and systems
 - Performance analysis, evaluation, optimization and prediction
 - Scientific workflow scheduling
 - Data visualization
 - On-demand computing
 - Virtualization supporting computations
 - Volunteer computing
 - Scaling applications from small-scale to exa-scale (and back)
 - Big data real-time computing / analytics

- Economic, business and ROI models for large-scale applications
- Emerging technologies for scalable computing
 - Cloud / Fog / Dew computing architectures, models, algorithms and applications
 - High performance computing in Cloud / Fog / Dew
 - Green computing in Cloud / Fog / Dew
 - Performance, capacity management and monitoring of Cloud / Fog / Dew configuration
 - Cloud / Fog / Dew application scalability and availability
 - Big Data cloud services
 - Architectures for large-scale computations (GPUs, accelerators, quantum systems, federated systems, etc.)
 - Self* and autonomous computational / storage systems

TECHNICAL SESSION CHAIRS

- **Ganzha, Maria**, Warsaw University of Technology, Poland
- **Gusev, Marjan**, University Sts Cyril and Methodius, Macedonia
- **Paprzycki, Marcin**, Systems Research Institute Polish Academy of Sciences, Poland
- **Petcu, Dana**, West University of Timisoara, Romania
- **Ristov, Sashko**, University of Innsbruck, Austria

PROGRAM COMMITTEE

- **Barbosa, Jorge**, University of Porto, Portugal
- **D’Ambra, Pasqua**, IAC-CNR, Italy
- **Gordon, Minor**, Software development consultant, United States
- **Gravvanis, George**, Democritus University of Thrace, Greece
- **Grosu, Daniel**, Wayne State University, United States
- **Holmes, Violeta**, The University of Huddersfield, United Kingdom
- **Kimovski, Dragi**, University of Klagenfurt, Austria
- **Margaritis, Konstantinos G.**, University of Macedonia, Greece
- **Nosovic, Novica**, Faculty of Electrical Engineering, University of Sarajevo, Bosnia and Herzegovina
- **Roszczyk, Radosław**, Warsaw University of Technology, Poland
- **Saurabh, Nishant**, University of Klagenfurt, Austria
- **Schikuta, Erich**, University of Vienna, Austria

- **Schreiner, Wolfgang**, Johannes Kepler University Linz, Austria
- **Shen, Hong**, University of Adelaide, Australia
- **Tudruj, Marek**, Inst. of Comp. Science Polish Academy of Sciences/Polish-Japanese Institute of Information Technology, Poland

Khaos: Dynamically Optimizing Checkpointing for Dependable Distributed Stream Processing

Morgan K. Geldenhuys*, Benjamin J. J. Pfister*, Dominik Scheinert*, Lauritz Thamsen[‡], and Odej Kao*
 Technische Universität Berlin, Germany, {firstname.lastname}@tu-berlin.de
[‡] University of Glasgow, United Kingdom, lauritz.thamsen@glasgow.ac.uk

Abstract—Distributed Stream Processing systems are becoming an increasingly essential part of Big Data processing platforms as users grow ever more reliant on their ability to provide fast access to new results. As such, making timely decisions based on these results is dependent on a system’s ability to tolerate failure. Typically, these systems achieve fault tolerance and the ability to recover automatically from partial failures by implementing checkpoint and rollback recovery. However, owing to the statistical probability of partial failures occurring in these distributed environments and the variability of workloads upon which jobs are expected to operate, static configurations will often not meet Quality of Service constraints with low overhead.

In this paper we present Khaos, a new approach which utilizes the parallel processing capabilities of cloud orchestration technologies for the automatic runtime optimization of fault tolerance configurations in Distributed Stream Processing jobs. Our approach employs three subsequent phases which borrows from the principles of Chaos Engineering: establish the steady-state processing conditions, conduct experiments to better understand how the system performs under failure, and use this knowledge to continuously minimize Quality of Service violations. We implemented Khaos prototypically together with Apache Flink and demonstrate its usefulness experimentally.

Index Terms—Distributed Stream Processing, Chaos Engineering, Quality of Service, Cloud, Parallel Profiling, QoS Modeling, Runtime Optimization

I. INTRODUCTION

WITH the growing necessity to quickly process large volumes of unbounded data, Distributed Stream Processing (DSP) systems are becoming an increasingly essential part of data processing environments. It is here where events must traverse a graph of streaming operators to allow for the extraction of results, which are at their most valuable closest to the time of data arrival. Data streams are continuously generated in a variety of contexts such as sensors in IoT network, network monitoring, financial fraud detection, click stream analytics, spam filtering, and social media [1], [2]. DSP systems are not only required to offer near to real-time processing latencies at high throughput rates, but also to recover from the various types of failures that inevitably occur in these environments.

DSP jobs are, in principle, required to operate indefinitely on unbounded streams of continuous data and exhibit heterogeneous modes of failure as they continue to run over long periods of time [3]. Consequently, DSP systems such as Apache Storm [4], Apache Spark [5], or Apache Flink [6] feature high availability modes and fault tolerance mechanisms that allow for results to be consistent in the presence of

partial failures. However, the complexity with which these systems are composed makes estimating how a system will perform through manual manipulation of the configuration sets a hard problem to solve. This is complicated by the fact that DSP jobs operating in an environment where streaming workloads change over time will make any static configuration selections obsolete in short order. This is especially relevant when considering critical jobs where the presence of Quality of Service (QoS) constraints dictate the minimum level of performance that is to be expected. It is therefore essential when optimizing the fault tolerance mechanism of DSP jobs to not only understand how configuration impacts upon recovery times as well as end-to-end processing latencies, but to ensure that the system is capable of reacting to changing workloads.

Checkpointing mechanisms are among the most popular and effective techniques for achieving fault tolerance in real world processing systems and consequently a number of methods for auto-configuration of checkpointing have been proposed. Most of them try to optimize the checkpoint interval by means of predicting or utilizing failure rates [7], the Mean Time To Failure (MTTF) [8]–[10], or recovery times [11], whereas some approaches even employ advanced multi-level checkpointing [12]–[17]. Yet, the majority of such methods either assume static workloads, consider solely offline optimization, or are primarily designed for high-performance computing (HPC) environments, which renders them not suitable for real-world DSP systems. Therefore, a workload-adaptive DSP configuration optimization approach, paired with a systematically attempt to evaluating DSP failure recovery performance executing in production-like environments, is needed.

In this paper we present Khaos, a novel approach for the automatic runtime optimization of DSP fault tolerance configurations. Borrowing from the principles of Chaos Engineering [18], Khaos achieves this by executing a three phase plan: Firstly, establishing the steady-state by recording and then analyzing the streaming workload of a targeted DSP job to identify interesting points for failure injection; Secondly, by taking advantage of container orchestration cloud technologies to replicate multiple pipelines in parallel where the workload is replayed, and then utilizing fine-grained failure injection together with an anomaly detector trained on normal pipeline executions to measure recovery times across a range of configuration settings and throughput rates; and thirdly, by taking the information gathered during profiling to train analytical models to monitor for when recovery times and latencies

would exceed user-defined QoS constraints and automatically optimize for better fault tolerance configurations at runtime. To determine if violations should be acted upon immediately or deferred for later, Khaos makes use of Time Series Forecasting (TSF) to predict future workloads. Khaos is applicable for users willing to initially accept an increased level of resource utilization to ensure optimized execution over the longer term.

The remainder of the paper is structured as follows: Section II explores the related work regarding DSP systems and their fault tolerance mechanisms as well as adaptive checkpointing schemes. Section III presents our approach to automatically optimizing the fault tolerance mechanism of targeted DSP jobs operating on variable workloads; Section IV describes our evaluation through performing two experiments; and Section V summarizes our findings.

II. RELATED WORK

In this section we examine how fault tolerance is handled in three popular DSP systems as well as work most related to our own, aimed at adaptive checkpointing and failure injection.

A. Distributed Stream Processing Systems

In DSP, checkpointing is the most popular fault tolerance mechanism for real world systems. One of the first widely used large-scale DSP systems is Apache Storm [4], which guarantees that in the event of failure, messages are re-processed by using a mechanism of upstream operator backup and message acknowledgements. Although such a mechanism does guarantee messages are processed at least once, it also results in duplicate messages passing through the system and, therefore, falls short of the exactly-once processing guarantees needed by many modern DSP jobs.

Apache Spark Streaming [5], on the other hand, is designed around the idea of micro-batching where messages are grouped together in an attempt to overcome the overhead caused by message-level synchronization. Here, micro-batches either succeed or are recomputed when failures occur. It uses periodic checkpointing to truncate the RDD lineage graph and save both metadata and data to reliable storage. This technique allows for exactly-once processing of messages.

Apache Flink [6] is a well-known DSP system, which likewise provides exactly-once processing guarantees through periodically creating and saving a distributed snapshot of the global state [19]. It achieves this by passing streaming barriers through the execution graph from source to sink operators. At the arrival of a barrier, each operator performs a checkpoint of the local state and then passes the barrier on to all output edges. A checkpoint is considered complete when all operators have completed their individual checkpoints.

B. Adaptive Checkpointing

A number of approaches have been proposed that optimize the fault tolerance configuration parameters by finding an optimal checkpoint interval (CI) to improve performance. Our previous work [11] focused on predicting recovery times

and then optimizing the CI with regards to a single user-defined QoS constraint. However, this approach is aimed at scenarios where jobs process a static workload, i.e. throughput does not change over time. In the closely related area of research, we published an approach which uses *times series forecasting* to optimize the resource utilization of DSP jobs executing in environments where the workload is expected to change over time [20]. A group of approaches focuses on determining the mean time to failure (MTTF) of cluster nodes and then adaptively fitting a CI that minimizes the time lost due to failure [8]–[10]. These approaches, however, are more appropriate to high-performance computing (HPC) clusters and batch processing jobs as they rely on jobs having a finite execution time as part of their calculations. Specific to DSP systems, [7] incorporates failure rates in an attempt to fit the CI based on the MTTF. However, unlike Khaos, optimizations are not performed at runtime and therefore dynamic workloads are not taken into account. Additionally, it does not incorporate the total time needed to recover to processing events at the latest timestamp nor consider any user-defined QoS constraints for its optimization step.

Other approaches have been proposed using multi-level checkpointing schemes to resolve the issue of checkpoint/recovery overhead [12]–[17]. Different checkpointing levels are used, which in turn are more flexible than traditional single-level schemes as it can consider multiple-failure types with each having a different checkpoint/recovery cost associated. Likewise, differing checkpoint levels can be associated with different storage types, which is usually not possible with single-level checkpoints. In [21], a two-level checkpointing model is proposed: checkpoint level 1 deals with errors with low checkpoint/recovery overheads such as transient memory errors, and checkpoint level 2 deals with hardware crashes such as node failures. These approaches are specific to HPC environments and need adaption before vendors can consider implementing them in DSP systems.

C. Failure injection

Here we present approaches which use failure injection as a means of gaining a greater insight into how systems perform when things go wrong. Failure injection for distributed systems is realized in [22] using virtualization at a low level. Both machines and networks are virtualized, and a failure injection is performed by uncontrolled shutdown of machines. The authors whole approach is accurately described as an emulation platform. In contrast, Khaos utilizes modern container orchestration technologies for deploying DSP pipelines and thus eases the emulation furthermore. In another work, fault injection is used in [23] to assess the effectiveness of partial fault tolerance techniques in DSP applications. The authors identify four metrics that can be used to evaluate the impact of faults in different stream operators with respect to predictability and availability. To reduce the number of required fault injection targets when evaluating a target application, their framework pre-analyzes the data flow graph. An approach that combines fault injection and data analytics is motivated in [24]. Here,

a database of failures is populated during a profiling phase, and queried during execution of production DSP jobs. The database is specifically used to help identify root causes of failures observed by providing injected faults that generated similar processing flows.

Netflix injects failures into its production system using its Chaos Automation Platform (ChAP) and Failure Injection Testing (FIT) tools [25]. ChAP performs Chaos experiments on a small, randomly selected percentage of live traffic. ChAP deploys two scaled-down clusters of the Netflix service, one to serve as the performance baseline, the other to serve as a canary group. FIT injects either error responses or increases latency to simulate failure scenarios [26] for the canary group and the results are compared against the baseline. Results of the Chaos experiment are evaluated by performing anomaly comparisons to detect statistically significant deviations between the normal state and Chaos experiment [25].

III. APPROACH

This section describes in detail the various phases of our approach Khaos, right after a general overview.

A. Idea Overview

The configuration of a fault tolerance mechanism has a direct impact both on the performance and availability of any running DSP job. Khaos borrows from the principles of Chaos Engineering and takes advantage of the massively parallelizable capabilities of container orchestration cloud technologies to provide an automated runtime approach for tuning the fault tolerance configuration of targeted DSP jobs. It achieves this by first conducting parallel profiling runs where failures are injected into short-lived profiling jobs, each testing a variation of the configurations, and gathering metrics related to the latencies and recovery times. These results are then used to train two multivariate runtime models that, when combined, provide a mechanism whereby user-defined performance and availability constraints are monitored for violations. Should this occur, the system can be automatically reconfigured to provide the best trade-off between constraints.

The cost of reconfiguration should likewise be taken into account, as it requires a full restart of the job with several current DSP implementations, albeit without having to reprocess any messages that might have accumulated during the downtime. This is because controlled restarts perform a system save immediately before making the change and therefore no consumer lag can build up. In order to do this we make use of TSF to determine if a reconfiguration should be performed at the current point in time, or if the decision can be deferred to the next optimization cycle. The logic behind this is that if the workload is expected to decrease substantially, then a reconfiguration is not necessary. Overall, such an optimization approach is imperative when operating in an environment where the workload changes dynamically over time and any static configurations are essentially made obsolete immediately. In order to achieve this, our approach is subdivided into

three distinct phases that are executed sequentially. Next we describe each of these phases and provide formal definitions.

B. Phase 1: Establishing the Steady State

The first phase focuses on capturing and establishing the steady state that describes job performance under failure-free conditions. A graphical overview of this phase can be seen in Figure 1(a). Consider a production-level DSP job a user would like to have adaptively optimized. The job is executing in a containerized environment, consuming a variable workload from a streaming platform, and metrics are gathered in a time series database. On initialization, Khaos connects to the streaming platform and begins recording the incoming event stream. It does this for a finite amount of time k as defined by the user and ideally should contain the maximum range of throughput rates as expected under normal processing loads across that period. For best results, DSP jobs processing a stationary data stream would best fit our approach. When considering any single timestamp t_i within this recording period, we can define the set of arriving events as $E^{(t_i)} = \{e_1^{(t_i)}, e_2^{(t_i)}, \dots, e_{n-1}^{(t_i)}, e_n^{(t_i)}\}$, where n denotes the number of arriving events in that timestamp. Consequently, the full set D of events arriving across all timestamps during the recording period is defined as

$$D = \{E^{(t_1)}, E^{(t_2)}, \dots, E^{(t_{k-1})}, E^{(t_k)}\}. \quad (1)$$

As Khaos is concerned with injecting failures into a running system to gain an understanding of how recovery times differ across various configurations, it needs to select a set of points over the full range of observed throughput rates where failures can be injected when the dataset is replayed. Therefore, a continuous function W of time t is extracted from D representing the workload over time and is defined by

$$W(t) = |E^{(t)}| \quad (2)$$

This function is analyzed to find a set of equidistantly spaced throughput rates between the minimum and maximum observed workloads and their corresponding timestamp values. Importantly an averaging window is used to smooth the workload function and remove outliers. Formally, we identify

$$\begin{aligned} t_{\min} &= \arg \min_{0 \leq j \leq k} W(j) \\ t_{\max} &= \arg \max_{0 \leq j \leq k} W(j) \end{aligned} \quad (3)$$

as the points in time with minimum and maximum workload. Subsequently, we find m equidistant points and arrive at a set F of timestamps representing the *failure points*, i.e.

$$\begin{aligned} F &= \{t_{\min}, t_{\min+h}, \dots, t_{\max-h}, t_{\max}\}, \\ h &= (t_{\max} - t_{\min}) / (m - 1). \end{aligned} \quad (4)$$

Since F is a set of timestamps, the corresponding set of throughput rates TR can be defined as

$$TR = \{W(f) | f \in F\}. \quad (5)$$

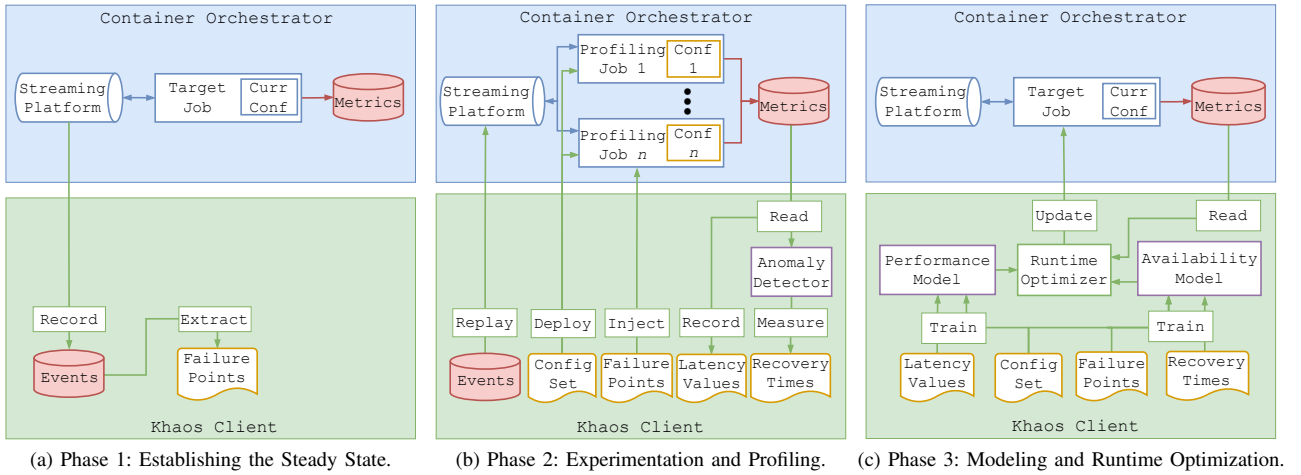


Fig. 1: Overview of Khaos.

On the conclusion of phase 1, the steady state of the production DSP job has been explored and the dataset D , failure points F and corresponding throughput rates TR are available for use in the next phase.

C. Phase 2: Experimentation and Profiling

The second phase focuses on performing experiments in an environment that is as similar as possible to production with the goal of gathering metrics data using the same DSP job but executed across a closed range of configuration settings. In order to do so we take advantage of container orchestration cloud technologies to rapidly replicate multiple profiling pipelines in parallel, record their operational latencies under differing throughput rates, and then use fine-grained failure injection together with an anomaly detector A to measure recovery times. Figure 1(b) provides a graphical overview of this phase and illustrates the interactions between components.

Concerning configuration of the fault tolerance mechanism, we are most concerned with the checkpoint interval, defined as the frequency with which the checkpoint process is initiated. It is by varying the CI that we are able to configure the fault tolerance mechanism, where higher values represent possible longer recovery times and, conversely, lower values represent faster recovery times. Varying the checkpoint interval will have an impact on both the overall performance and availability of the system. For the CI, we seek to investigate a range of z equidistantly spaced values given a minimum value and a maximum value. We effectively obtain a set of concrete configurations C with $z = |C|$ analogue to the procedure for F . Thus, we can test all configurations in C at the same time, i.e. arriving at as many deployments as configuration values.

After the parallel profiling jobs have been instantiated, each timestamp in the failure points F is registered with a built-in failure injector. This is so that when the timestamp, and therefore the associated throughput rate, is realized, Khaos will inject a failure concurrently into each one of the parallel deployments. Once this has been completed, Khaos will begin

replaying the dataset D at the same rate at which it was recorded. All parallel profiling jobs will read from the same source to which the data is being replayed. It is important to note that this is a separate messaging queue to the one being consumed by the production job. At the point before injecting the failure, an average latency measurement is taken for each of the parallel deployments forming the set L with

$$L = \{l_i^{(j)} | 1 \leq i \leq m, 1 \leq j \leq z\}, \quad (6)$$

where $l_i^{(j)}$ is the average latency measurement corresponding to the i -th failure injection into the j -th deployment. Although the full recorded dataset can be replayed, to reduce resource utilization during profiling, the user can specify a time interval where events just prior to and after the points of failure injection can be replayed thus limiting time spent in this phase. At the same time, performance metrics are gathered across all parallel deployments in order to measure recovery times. The metrics we are most concerned with include:

- **Input Throughput:** Measured in events per second, this value represents the sum of the events entering the source operators of the DSP job per second.
- **Average Consumer Lag:** Measured in number of events, this value represents the number of events accumulated in the messaging queue waiting to be consumed by the source operators of the DSP job.

In order to measure how long it takes for the DSP job to recover after experiencing a failure, the aforementioned metrics are used to train an anomaly detection algorithm on positive executions, i.e. let the function $s : X \rightarrow X$ perfectly represent the metrics data stream such that for any given data point $x \in X$ the prediction is always $s(x) = x$. Assuming most data collected in the recording period is normal, the algorithm can learn to detect deviations from the expected normal behaviors and therefore will report an anomalous behavior if a configurable threshold based on a window of

past errors is hit. Measuring the length of time the system was in an anomalous state is therefore equivalent to the recovery time. In order to accomplish this, we utilize an online ARIMA method motivated in [27] for our implementation.

It is important to note that in this context, the recovery time is not only referring to the time the system is in an inconsistent state before processing resumes. For systems employing checkpoint and rollback recovery strategies, processing will start at a previous saved offset and then attempt to catch up to the latest offset, all while new events are continuously arriving. It is this length of time, from when the failure occurs to the point at which processing is once again producing results at the latest offset, that we are interested in measuring, as it is a more accurate reflection of availability. Thus, we define the set of recovery times as measured by the anomaly detector as

$$R = \{r_i^{(j)} | 1 \leq i \leq m, 1 \leq j \leq z\}. \quad (7)$$

Consequently, the lengths of the observed recovery times in R are influenced by two main factors: the variable nature of the workload, i.e. the changing number of messages arriving per second over time; and the point at which the failure occurs relative to the next checkpoint completed successfully. As we intend to create a prediction model in the next phase using the recovery time observations, these factors need to be considered as they introduce variance influencing comparability, thus making our models unusable without further adaptations. Concerning the workload, we make the assumption that over these shorter time periods where recovery takes place, i.e. less than 15 minutes, the changing throughput rates would average out over time. Therefore, a constant workload can be assumed. However, the point at which the last checkpoint completed successfully will differ across failures and will directly impact the number of messages that need to be reprocessed, thus increasing or decreasing the recovery time. Therefore, for the purposes of our work, we only consider the worst-case scenario, i.e. we assume failures occur at the point right before the next checkpoint completes successfully. As such, for profiling runs, we measure the distance in time until the next checkpoint is scheduled to start, and inject the failure just prior to this point.

At the conclusion of the profiling runs, the parallel deployments are deleted and the resources are released, with the profiling sets C , TR , L , and R being passed into the third and final phase addressing modeling and runtime optimization.

D. Phase 3: Modeling and Runtime Optimization

The third and final phase is intended to execute indefinitely while continuously optimizing the targeted DSP job by monitoring for violations of two user-defined QoS constraints: l_{const} which defines an upper bound on the average end-to-end latency; and r_{const} which defines an upper bound on predicted recovery time. A violation of either constraint triggers a reconfiguration of the system where a new CI is chosen. Care must be taken when choosing a new CI value, as increasing the frequency with which checkpoints are performed will result

in better recovery times but could likewise negatively impact latencies and vice versa. Therefore, we formulate this as an optimization problem where the objective is to select a CI that minimizes for both performance and availability. A graphical representation can be seen in Figure 1(c).

In order to achieve this, we train two multiple regression models using the data gathered in the preceding two phases. The performance model M_L aims at finding a mapping such that $M_L : C, TR \rightarrow L$, whereas the recovery time model M_R is configured as $M_R : C, TR \rightarrow R$. Once both models have been trained with our observed values, metrics from the targeted DSP job are continuously gathered and evaluated. For performance violations, Khaos compares the current average end-to-end latency to the performance constraint l_{const} . As end-to-end latencies tend to be quite volatile, our previously fitted models likely contain noise for which we need to account when making actual predictions. Thus, we employ a correction approach for the prospective prediction error to localize predictions to the current cluster conditions. Khaos keeps track of the latency observations over the past k optimization iterations, averages across the k pairwise fractional differences given the current latency, and then uses this estimated rescaling factor p to rescale the predicted value obtained from our model. For recovery time violations, it determines the average throughput rate and together with the current CI uses M_R to predict the recovery time considering the worst case scenario.

However, reconfiguration is not without its own cost and requires a full restart of the job. Therefore, when violations of the constraints are detected, Khaos will determine whether or not a reconfiguration should be performed immediately or if this decision should be deferred until the next optimization cycle. In order to achieve this, a TSF model is trained on the incoming message rate and a multi-step ahead forecast is performed should a violation be detected. If the prospective incoming message rate is expected to decrease significantly, i.e. more than 10%, from the current point in time until the next optimization run is executed, then the decision to reconfigure can be delayed. Otherwise, the reconfiguration can proceed as per normal. The goal of reconfiguration is to select a CI value that results in the furthest distance from the two upper bounds that satisfies both. Formally, this multi-objective optimization problem can be formulated as

$$\begin{aligned} \min_C \quad & Q_R + Q_L^* + |Q_R - Q_L^*| \\ \text{s.t.} \quad & Q_R < r_{\text{const}}, \\ & Q_L^* < l_{\text{const}}, \\ & Q_R, Q_L^* > 0. \end{aligned} \quad (8)$$

Here, Q_R is the fraction $\frac{M_R(C, TR_{\text{avg}})}{r_{\text{const}}}$ between the prediction of the recovery time model and the corresponding constraint. The same applies to the latency model, except that Q_L^* describes the fraction after rescaling, i.e. $Q_L^* = p \cdot Q_L$. Given our configuration set C and the current average throughput rate TR_{avg} , we aim at finding a value for the CI that satisfies both objectives individually. If minimizing the above expression

finds a new value for the CI, which is predicted to be more performant, then the system is reconfigured should the expected workload not decrease and monitoring continues.

IV. EVALUATION

Now we show that Khaos is both practical and beneficial for DSP through experimentation. The prototype, data, and experiment artifacts can be found in the following repository¹.

A. Experimental Setup

Resource	Details
OS	Ubuntu 18.04.3
CPU	Quadcore Intel Xeon CPU E3-1230 V2 3.30GHz
Memory	16 GB RAM
Storage	3TB RAID0 (3x1TB disks, linux software RAID)
Network	1 GBit Ethernet NIC
Software	Java v1.11, Flink v1.12, Kafka v2.6, ZooKeeper v3.6, Docker v19.3, Kubernetes v1.18, HDFS v2.8, Redis v5.0, Prometheus v2.25

TABLE I: Cluster Specifications

Our experimental setup consisted of a co-located 50-node Kubernetes [28] and HDFS [29] cluster as well as a 3-node Apache Kafka² cluster configured to have 8 partitions and a replication factor of 3. Node specifications and software versions are summarized in Table I. A single switch connected all nodes. Each experiment consisted of a Kubernetes namespace containing: an Apache Flink³ native session cluster with all jobs set to have a parallelism of 4; and a single Prometheus⁴ time series database was used for the gathering of metrics. The Yahoo Streaming Benchmark (YSB) experiment additionally made use of a Redis⁵ database. Regarding end-to-end latency measurements, averages were taken over the 99th percentile in order to filter outliers during normal failure free operations. The timeout interval for Flink taskmanager nodes is 50s as per the default settings. Importantly, for both experiments QoS constraints for performance and availability were set at 1000ms for end-to-end latencies and 240s for recovery times respectively. Each experiment was conducted 5 times with the median selected for our results and discussion.

B. IoT Vehicles Experiment

We created a simulation that mapped the streets and intersections of an area of central Berlin, Germany. In this area a number of vehicles were generated travelling along various routes and providing an update message every 1 second. Each update message contained the vehicle_ID, vehicle_type, geolocation, speed, direction, and event_time. This IoT Vehicles streaming dataset was generated using Sumo [30] and the number of concurrent vehicles, i.e. the workload, is based on

¹<https://github.com/dos-group/khaos>

²<https://kafka.apache.org/>, Accessed: May 2022

³<https://flink.apache.org/>, Accessed: May 2022

⁴[https://prometheus.io.](https://prometheus.io/), Accessed: May 2022

⁵<https://redis.io/>, Accessed: May 2022

TABLE II: IoT Vehicles Experiment Results.

(a) Error Analysis.						
	Performance		Availability			
Avg. Percent Error	0.099		0.131			

(b) IoT Vehicles Experiment Results.						
Configuration	Khaos	10s	30s	60s	90s	120s
Avg. Latency (ms)	737	1086	729	796	697	692
Lat Violations (%)	0.087	0.153	0.062	0.110	0.073	0.060
Recovery Time (s)	2071	2757	1681	2064	2505	2904
Rec Violations (s)	197	1188	147	227	555	826

the TAPASCologne scenario⁶. For this experiment, a 7-day streaming dataset was generated using random seeds to create variability across the various days. Throughput rates over time can be seen in Figure 2(a).

A DSP job was created that processes the streaming vehicle data. It consisted of the following streaming operations: read an event from Kafka; deserialize the JSON string; filter update events not within a certain radius of a designated geo-point where vehicles are to be monitored; take a 3s sliding window with a slide of 1s where all update events are of the same vehicle ID and calculate the vehicle's average speed; generate a notification for vehicles that have exceeded the speed limit; enrich notification with vehicle type information from data stored in system memory and write it back out to Kafka.

C. YSB Experiment

This experiment is based on the Yahoo Streaming Benchmark⁷. It implements a simple streaming advertisement job where there are a number of advertising campaigns and a number of advertisements for each campaign. The authors of the benchmark created a Kafka Producer application that would generate a constant stream of events containing, among other things, an event_time, an event_type, and an ad_id. We created a generator that was combined with a click-through rate dataset⁸ to create the workload for this experiment. This workload can be in Figure 2(b).

The job of the benchmark is to read various JSON events from Kafka, identify relevant events, and store a windowed count of these events per campaign in Redis. The job consists of the following operations: read an event from Kafka; deserialize the JSON string; filter out irrelevant events (based on type field), take a projection of the relevant fields (ad_id and event_time), join each event by ad_id with its associated campaign_id stored in Redis; take a 10s windowed count of events per campaign and store each window in Redis along with a timestamp of when the window was last updated. For

⁶<https://sumo.dlr.de/docs/Data/Scenarios/TAPASCologne.html>; Accessed: May 2022

⁷<https://yahooeng.tumblr.com/post/135321837876/>

benchmarking-streaming-computation-engines-at, Accessed: May 2022

⁸<https://www.kaggle.com/c/avazu-ctr-prediction>, Accessed: May 2022

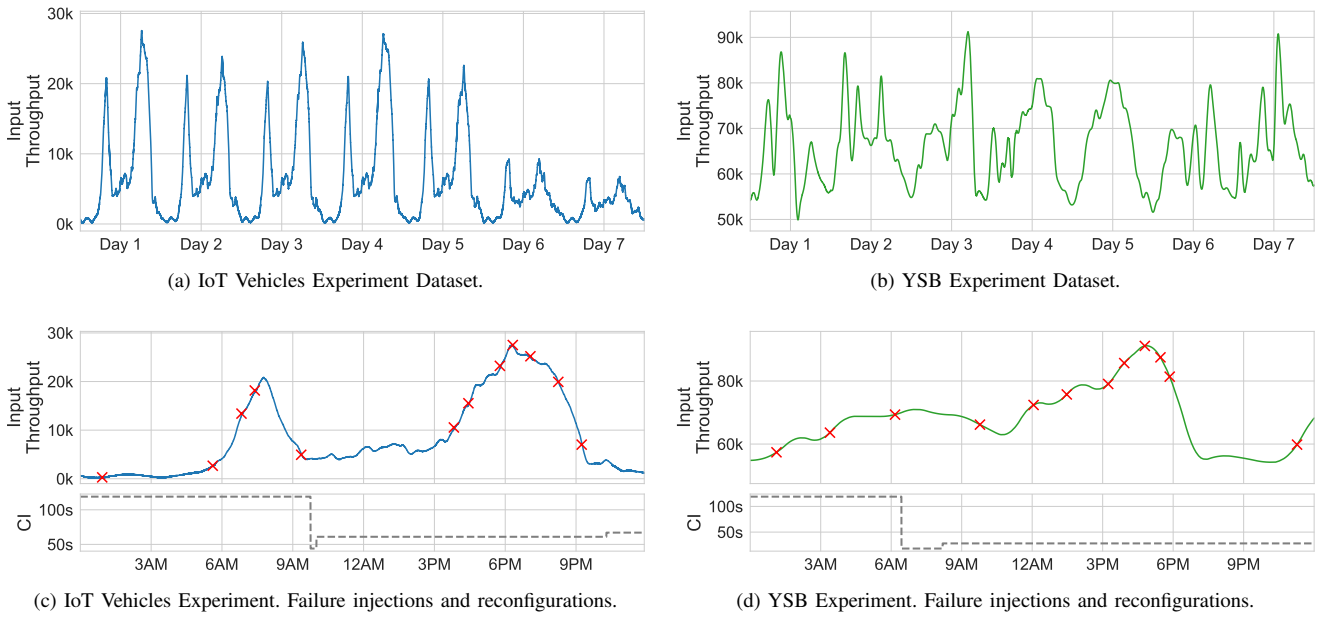


Fig. 2: Overview of datasets, their throughput rates, utilized failure injection points, and CI reconfigurations triggered by Khaos.

TABLE III: YSB Experiment Results.

(a) Error Analysis.

	Performance	Availability
Avg. Percent Error	0.122	0.0728

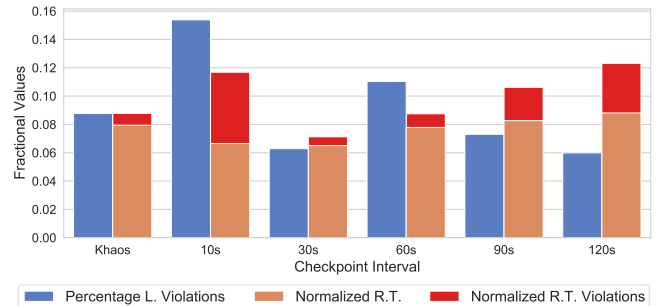
(b) YSB Experiment Results.

Configuration	Khaos	10s	30s	60s	90s	120s
Avg. Latency (ms)	653	691	660	637	576	527
Lat Violations (%)	0.059	0.061	0.069	0.058	0.033	0.024
Recovery Time (s)	2319	2182	2126	2548	3093	3532
Rec Violations (s)	9	117	32	77	401	764

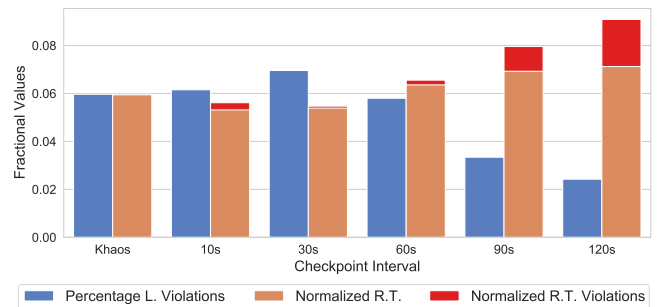
the purposes of our experiments, we modified the Flink benchmark by enabling checkpointing and replacing the handwritten windowing functionality with the default Flink implementation. Although doing so decreases the update frequency to the length of each window, results should be accurate and more interesting for our experiments due to the accumulated windowing operator state at each node.

D. Experimental Results & Discussion

The results of both experiments will now be presented and discussed in further detail. The effectiveness of our approach was evaluated against 5 baseline runs, which were executed in parallel to ensure cluster conditions were commensurate. For baseline runs, a range of static CI values were selected and streaming jobs started using these configurations. Each job consumed the same stream of events. Static CI values included 10, 30, 60, 90, and 120 seconds respectively. Over



(a) IoT Vehicles Experiment.



(b) YSB Experiment.

Fig. 3: Latency (L.) violations, normalized recovery times (R.T.), and normalized recovery time violations.

the course of each experiment, 12 failures were injected at similar times across all jobs. Failure selection was based on different throughput levels which can be seen for both experiments in Figure 2(c) and Figure 2(d). To ensure results were comparable, failures were injected at the end of the CI. This is inline with our assumption of the worst case scenario, i.e. a failure occurs just before the next checkpoint completes successfully and therefore maximizing recovery times.

Concerning the predictive models trained as part of the profiling phase for each experiment, the results of a post execution error analysis can be seen in Table II(a) and Table III(a). These results measured the average percent error between predicted and actual observations for both latencies and recovery times across the full duration of the experiments. Latency values were collected as part of the optimization loop and recovery times were measured when failures were injected. The results show that on average latency predictions were within 9% and 12% of expected and recovery time predictions were within 13% and 7% of expected. Considering the presence of this error within our predictions, we conclude that it was accurate enough for the optimization step to make good CI selections. We previously described our failure injection procedure highlighted in Figure 2(c) and Figure 2(d), we will now describe the bottom section of this figure. Associated results for these experiments can be seen in Table II(b) and Table III(b). During optimization, Khaos triggers reconfigurations of the CI in order to account for changes in latency and prospective recovery times. We observed in total three reconfigurations for the IoT Vehicles Experiment, and two reconfigurations for the YSB Experiment. It can be seen that the CI is often set to a lower value with increasing throughput rates. This is expected, as formulated recovery time constraints can likely not be fulfilled when maintaining a high CI value. In general, the few reconfigurations result from our method requiring one of the constraints to be fulfilled in order to conduct an optimization step. Consequently, reconfigurations are applied sparsely, and CI configurations in-between observed violations are not guaranteed to be optimized if both constraints are intermittently jointly violated. An example for this can be seen in Figure 3(a) with recovery time violations for Khaos, indicating that CI updates were aborted at some point.

Regarding the violation of formulated performance and availability constraints, we report in Figure 3 the fraction of time the latency constraint was not fulfilled, and normalize the measured recovery times accordingly such that the resulting bars are aligned for Khaos. This allows us to better assess the relation of both objectives for the utilized static CI values, i.e. our baselines. It can be clearly showed that the CI has an impact on latencies, i.e. with more frequent checkpoints being performed, latencies are higher and vice versa with recovery times. This often leads to violations of the respective formulated constraints. In our conducted experiments, this is less problematic for smaller CI values, which is also what Khaos often defaults to (see Figure 2). An exception to this is the 10s CI illustrated in Figure 3(a), which performs poorly due to a failure injection during catch-up from a previous failure.

However, while certain static CI configurations show comparably good results, this is restricted to our exemplary streaming jobs only, i.e. the discovered CIs are not a general solution. Based on the results presented in Table II(b) and Table III(b), we can surmise that Khaos produces better average latencies overall than the high frequency CI configurations. Likewise the percentage of latency violations were commensurate with these configurations. At the same time, Khaos produced fewer recovery time violations indicating that it provides a balance of both within the user-defined constraints. It is important to note that while individual static configurations might marginally outperform Khaos for a specific job and workload, the benefits of such a selection will not generalize if the jobs and/or workloads change.

V. CONCLUSION

In this paper we presented Khaos, an approach which borrows from the principles of Chaos Engineering to allow for the automatic runtime optimization of DSP fault tolerance configurations. It makes use of parallel profiling runs and failure injection to capture metrics, which are in turn used to model the performance and availability of targeted DSP jobs executing on variable workloads. It does this in order to monitor for runtime violations of user-defined QoS constraints and, should a violation be detected, can search for and select near-optimal CI configurations, which provide a balance of both. Through our experiments we showed that Khaos is able to optimize the CI configuration variable in order to minimize both latency and recovery time violations while outperforming most static configurations. For future work we intend to investigate the feasibility of a continuous optimization routine which takes periods of low utilization into consideration.

ACKNOWLEDGMENT

This work has been supported through grants by the German Ministry for Education and Research (BMBF) as BIFOLD (funding mark 01IS18025A) and WaterGridSense 4.0 (funding mark 02WIK1475D).

REFERENCES

- [1] H. Isah, T. Abughofa, S. Mahfuz, D. Ajerla, F. H. Zulkernine, and S. Khan, "A survey of distributed data stream processing frameworks," *IEEE Access*, vol. 7, 2019.
- [2] H. Nasiri, S. Nasehi, and M. Goudarzi, "Evaluation of distributed stream processing frameworks for iot applications in smart cities," *J. Big Data*, vol. 6, p. 52, 2019.
- [3] H. Li, J. Wu, Z. Jiang, X. Li, X. Wei, and Y. Zhuang, "Integrated recovery and task allocation for stream processing," in *IPCCC*. IEEE Computer Society, 2017.
- [4] A. Toshniwal, S. Taneja, A. Shukla, K. Ramasamy, J. M. Patel, S. Kulkarni, J. Jackson, K. Gade, M. Fu, J. Donham, N. Bhagat, S. Mittal, and D. V. Ryaboy, "Storm@twitter," in *MOD*, C. E. Dyreson, F. Li, and M. T. Özsu, Eds. ACM, 2014.
- [5] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica, "Spark: Cluster computing with working sets," in *HotCloud*, E. M. Nahum and D. Xu, Eds. USENIX Association, 2010.
- [6] P. Carbone, A. Katsifodimos, S. Ewen, V. Markl, S. Haridi, and K. Tzoumas, "Apache flink™: Stream and batch processing in a single engine," *IEEE Data Eng. Bull.*, vol. 38, no. 4, pp. 28–38, 2015.
- [7] S. Jayasekara, A. Harwood, and S. Karunasekera, "A utilization model for optimization of checkpoint intervals in distributed stream processing systems," *Future Gener. Comput. Syst.*, vol. 110, pp. 68–79, 2020.

- [8] J. W. Young, "A first order approximation to the optimum checkpoint interval," *Commun. ACM*, vol. 17, pp. 530–531, 1974.
- [9] J. Daly, "A model for predicting the optimum checkpoint interval for restart dumps," in *ICCS*, ser. LNCS, P. M. A. Sloot, D. Abramson, A. V. Bogdanov, J. J. Dongarra, A. Y. Zomaya, and Y. E. Gorbachev, Eds., vol. 2660. Springer, 2003.
- [10] J. T. Daly, "A higher order estimate of the optimum checkpoint interval for restart dumps," *Future Gener. Comput. Syst.*, vol. 22, no. 3, 2006.
- [11] M. K. Geldenhuys, L. Thamsen, and O. Kao, "Chiron: Optimizing fault tolerance in qos-aware distributed stream processing jobs," in *IEEE International Conference on Big Data, Big Data 2020, Atlanta, GA, USA, December 10-13, 2020*. IEEE, 2020, pp. 434–440.
- [12] L. A. Bautista-Gomez, A. Nukada, N. Maruyama, F. Cappello, and S. Matsuoka, "Low-overhead diskless checkpoint for hybrid computing systems," in *HiPC*. IEEE Computer Society, 2010.
- [13] L. A. Bautista-Gomez, N. Maruyama, F. Cappello, and S. Matsuoka, "Distributed diskless checkpoint for large scale systems," in *CCGrid*. IEEE Computer Society, 2010.
- [14] H. Li, L. Pang, and Z. Wang, "Two-level incremental checkpoint recovery scheme for reducing system total overheads," *PLoS ONE*, vol. 9, 2014.
- [15] A. Moody, G. Bronevetsky, K. Mohror, and B. R. de Supinski, "Design, modeling, and evaluation of a scalable multi-level checkpointing system," in *SC*. IEEE, 2010.
- [16] L. A. Bautista-Gomez, S. Tsuboi, D. Komatitsch, F. Cappello, N. Maruyama, and S. Matsuoka, "FTI: high performance fault tolerance interface for hybrid systems," in *SC*, S. A. Lathrop, J. Costa, and W. Kramer, Eds. ACM, 2011.
- [17] A. Kulkarni, A. Manzanares, L. Ionkov, M. Lang, and A. Lumsdaine, "The design and implementation of a multi-level content-addressable checkpoint file system," in *HiPC*. IEEE Computer Society, 2012.
- [18] A. Basiri, N. Behnam, R. de Rooij, L. Hochstein, L. Kosewski, J. Reynolds, and C. Rosenthal, "Chaos engineering," *IEEE Softw.*, vol. 33, no. 3, pp. 35–41, 2016.
- [19] P. Carbone, G. Fóra, S. Ewen, S. Haridi, and K. Tzoumas, "Lightweight asynchronous snapshots for distributed dataflows," *CoRR*, vol. abs/1506.08603, 2015.
- [20] M. K. Geldenhuys, D. Scheinert, O. Kao, and L. Thamsen, "Phoebe: Qos-aware distributed stream processing through anticipating dynamic workloads," *ICWS*, 2022.
- [21] S. Di, Y. Robert, F. Vivien, and F. Cappello, "Toward an optimal online checkpoint solution under a two-level HPC checkpoint model," *IEEE Trans. Parallel Distributed Syst.*, vol. 28, no. 1, pp. 244–259, 2017.
- [22] T. Hérault, T. Largillier, S. Peyronnet, B. Quétier, F. Cappello, and M. Jan, "High accuracy failure injection in parallel and distributed systems using virtualization," in *CF*, G. Johnson, C. Trinitis, G. Gaydadjiev, and A. V. Veidenbaum, Eds. ACM, 2009.
- [23] G. Jacques-Silva, B. Gedik, H. Andrade, K. Wu, and R. K. Iyer, "Fault injection-based assessment of partial fault tolerance in stream processing applications," in *DEBS*, D. M. Eysers, O. Etzion, A. Gal, S. B. Zdonik, and P. Vincent, Eds. ACM, 2011.
- [24] C. Pham, L. Wang, B. Tak, S. Baset, C. Tang, Z. T. Kalbarczyk, and R. K. Iyer, "Failure diagnosis for distributed systems using targeted fault injection," *IEEE Trans. Parallel Distributed Syst.*, vol. 28, no. 2, pp. 503–516, 2017.
- [25] A. Basiri, L. Hochstein, N. Jones, and H. Tucker, "Automating chaos experiments in production," in *ICSE*, H. Sharp and M. Whalen, Eds. IEEE / ACM, 2019.
- [26] A. Blohowiak, A. Basiri, L. Hochstein, and C. Rosenthal, "A platform for automating chaos experiments," in *ISSRE*. IEEE Computer Society, 2016.
- [27] F. Schmidt, F. Suri-Payer, A. Gulenko, M. Wallschläger, A. Acker, and O. Kao, "Unsupervised anomaly event detection for VNF service monitoring using multivariate online arima," in *CloudCom*. IEEE Computer Society, 2018.
- [28] A. Verma, L. Pedrosa, M. Korupolu, D. Oppenheimer, E. Tune, and J. Wilkes, "Large-scale cluster management at google with borg," in *EuroSys*, L. Réveillère, T. Harris, and M. Herlihy, Eds. ACM, 2015.
- [29] K. Shvachko, H. Kuang, S. Radia, and R. Chansler, "The hadoop distributed file system," in *MSSST*, M. G. Khatib, X. He, and M. Factor, Eds. IEEE Computer Society, 2010.
- [30] P. Á. López, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. WieBner, "Microscopic traffic simulation using SUMO," in *ITSC*, W. Zhang, A. M. Bayen, J. J. S. Medina, and M. J. Barth, Eds. IEEE, 2018.

Increasing data availability and fault tolerance for decentralized collaborative data-sharing systems

Kamil Jarosz*, Łukasz Opiola†, Łukasz Dutka†, Renata G. Słota*, and Jacek Kitowski*†

* AGH University of Science and Technology,
 Faculty of Computer Science, Electronics and Telecommunications,
 Institute of Computer Science, Krakow, Poland

Emails: {kamil.jarosz, renata.slota, jacek.kitowski}@agh.edu.pl

† Academic Computer Centre CYFRONET AGH, Krakow, Poland

Emails: lopiola@agh.edu.pl, lukasz.dutka@cyfronet.pl

Abstract—In order to realize collaboration on a global scale, academic research requires often large quantities of data to be shared between geographically dispersed organizations. The requirement to protect and govern data in a network of loosely coupled, autonomous institutions is an incentive for decentralized solutions, where the participants are in full control of their data without trusting a third-party provider to store and process the data. In order to increase data availability and fault tolerance in decentralized collaborative systems, we propose a layer, which is based on replication and decentralized authority over the data. The solution consists of an idea of peer-sets, which are groups of peers implementing collective data management, a consensus protocol which synchronizes a distributed ledger between peers, and an atomic commitment protocol used to implement optional two-way references between documents. This architecture may be utilized in various decentralized collaborative data-sharing systems, such as Onedata.

I. INTRODUCTION

DEMAND for global data access and data availability is growing due to increasing globalization and scale. It is especially evident in research, where often large quantities of data need to be shared upon request easily between geographically dispersed groups of people. High data availability and safety is usually a requisite in order to perform large scale computations. For instance, the eXtreme-DataCloud project [1] developed smart orchestration tools and building blocks of software, which provide storage federation with transparent or quasi-transparent data access for large and geographically distributed datasets. This project was dedicated to prominent research communities from various domains: LifeScience, Biodiversity, Clinical Research, Astrophysics, High Energy Physics, and Photon Science.

Collaboration between researchers may be backed by scientific organizations, which are autonomous and often do not have any written agreements between them. Each organization manages its own data, which may reside on various storage systems in different clouds. The requirement to protect and govern their data in a network of loosely coupled, autonomous institutions makes it difficult to constrain data sharing software to be centralized. Instead, decentralized solutions are more suitable in such cases, because the participants are in full control of their data and they may decide what is shared to whom, without trusting a third-party provider to store and

process the data. Examples include ScienceMesh [2], which tries to connect existing storage systems and services using a common API, or Onedata [3], which delivers a service allowing to create a global network of independent storage providers.

High data availability in collaborative systems is crucial, but is also more difficult to accomplish in decentralized environments. When a party experiences a failure, all of its data becomes unavailable to others, whom parts of the data have been shared with. The problem is more complex, because not only may collaborators want to access the shared data, but also modify it—all of it during the outage of the original organization’s infrastructure. These scenarios show a demand not only for decentralized architecture of data sharing systems, but also for decentralized authority over shared data, i.e. a possibility to modify it collectively by each one of its owners—collaboration between organizations often results in a joint work, and it is not uncommon for people to be parts of multiple organizations at once. In this paper, we propose an architecture of a layer of data-sharing systems, which ensures high availability and fault tolerance, and is based on a global, decentralized network of peers. The novelty of our proposal lies in dividing the global network into smaller peer-sets, which allow unrestricted global collaboration, at the same time limiting the flow of information between peers to the required minimum.

II. RELATED WORK

The concept of decentralized collaboration and data-sharing has been studied for years. Onedata is an important example, the goal of which is to achieve collaboration in global networks of autonomous organizations. It provides a service called Onezone, which implements a common layer between different storage providers and enables universal user authentication [4]. Other notable work include ScienceMesh [5], which provides an ecosystem—common interfaces and tools in order to connect existing heterogeneous services. Its goal is to allow collaboration using, for instance, share-with flows or data synchronization. It is relatively simple and does not implement more advanced solutions, such as complex organizational structures.

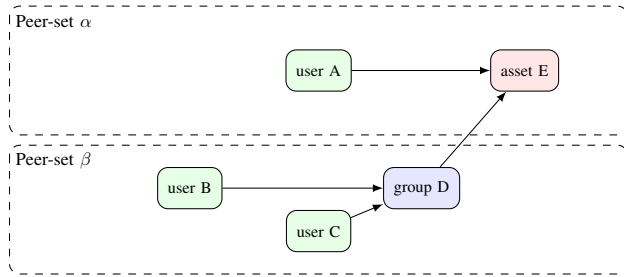


Fig. 1. Exemplary organizational structure spanned across two peer-sets.

It is possible to create a global network of independent peers using various DHT implementations, but they have inherent problems with security in trustless environments. Blockchain [6], on the other hand, ensures Byzantine Fault Tolerance and can be used both for decentralized storage [7], and for decentralized collaboration [8]. Blockchain is problematic due to its publicness—in situations where data is to be kept private its use is limited. In order to constrain possible participants and limit public access, permissioned blockchains may be used. However, they require a central authority responsible for authorization which undermines their decentralization. Irrespectively of the type of blockchain used, the primary issue is related to the commitment of organizations to handle a global chain and store data accessible to other blockchain participants regardless of their willingness to collaborate.

III. OUR PROPOSAL

We propose a decentralized architecture of a layer of data-sharing systems with goals of increasing data availability and fault tolerance. It is achieved by utilizing decentralized authority over the data and replicating it over several peers. This section also includes a discussion on protocols and structures of the data used. We target the metadata used to describe organizational structures, users, groups, etc., but the architecture overall may be used for more generic document-like data with optional two-way references between them.

A. Architecture

We assume that the system stores and allows access to “objects”, which represent arbitrary metadata. Objects may have relations between them, which are represented as two-way references. For instance, in case of organizational structures, an object may represent a user, a group, or an asset. Relations may represent memberships, in case of users and groups, or access grants, in case of assets (cf. Fig. 1).

Objects are decentralized by nature—each object is owned by some organization—a peer. For instance, an object representing a user may be owned by its university (an organization), along with his groups. Collaboration between different organizations happens when users belonging to each have common relations, e.g. they are both members of the same group.

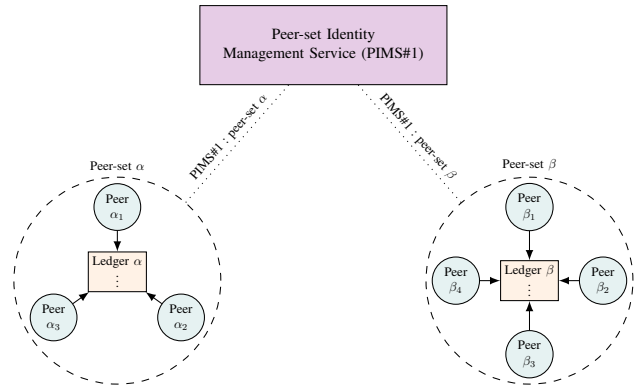


Fig. 2. Exemplary logical structure of peer-sets: each peer-set stores its own ledger, whereas PIMSeS resolve peers inside peer-sets.

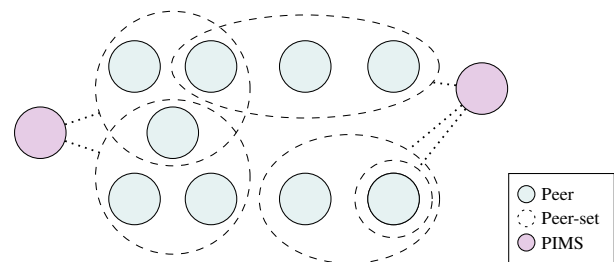


Fig. 3. Exemplary network consisting of PIMSeS and peers divided into peer-sets; one peer may belong to an arbitrary number of peer-sets, peer-sets may consist of arbitrary numbers of peers.

We propose an idea of a “peer-set”—a group of peers which collectively manages a set of objects (cf. Fig. 2 and 3). This concept increases data availability and fault-tolerance, because objects are not managed in a centralized manner, but rather are managed collectively by multiple peers. This way, users with multiple affiliations may be naturally managed by a peer-set consisting of organizations they are part of. Groups and assets may be spanned across organizations which support the work they are used for, or they may be backed up to other peers provided by the organization. Every peer-set is a closed group, which may be created by a user, given permissions from each peer. We assume that every member of a peer-set is not malicious, and the peer-set creator trusts each peer to store and manage the data, thus we do not consider Byzantine faults.

In order to increase flexibility, our proposal also includes mutability of peer-sets. This allows users to dynamically add or remove a peer from the peer-set, which improves data availability.

B. Peer-set ledger structure and consensus protocol

A fundamental problem in distributed systems is coordinating multiple peers to agree on some common value, i.e. reach *consensus*. Consensus protocols—created to solve this problem—have to be fault tolerant, as an arbitrary set of participating peers may fail at any moment. The conventional consensus protocols include Paxos [9] along with its modifica-

TABLE I
PEER-SET α LEDGER CONTENTS

#	Peer-set α
4	Add relation “group D from β ” \rightarrow “asset E”
3	Add relation “user A” \rightarrow “asset E”
2	Add object “asset E”
1	Add object “user A”

TABLE II
PEER-SET β LEDGER CONTENTS

#	Peer-set β
6	Add relation “group D” \rightarrow “asset E from α ”
5	Add relation “user C” \rightarrow “group D”
4	Add relation “user B” \rightarrow “group D”
3	Add object “group D”
2	Add object “user C”
1	Add object “user B”

tions, such as Multi-Paxos [10] or EPaxos [11]. There are also alternatives, such as Raft [12] or Zab [13]. Some consensus protocols may also be categorized as Byzantine fault-tolerant, which imposes a weaker condition on the type of faults that may happen—instead of only assuming crashes, they allow an arbitrary failure, including malicious misbehaving, such as forging counterfeited messages. Such consensus protocols include proof-of-work or proof-of-stake [14], both of which are used mainly in public blockchain networks [6]. There are also BFT versions of consensus protocols used outside of blockchain, such as PBFT [15].

In case of peer-sets, all peers inside them need to synchronize and decide on common state of the managed objects. Thus, we propose a peer-set history structure based on a distributed ledger, which consists of a list of incremental changes to the objects. The ledger is synchronized between peers using a consensus protocol based on examples mentioned above. Every peer may propose a change which extends the ledger by a new entry. Tables I and II depict exemplary ledger contents of respectively peer-set α and peer-set β , which are consistent with the state shown in figure 1. Solutions such as QLDB [16] may be used as an implementation for the ledger.

Our approach assumes not only data replication, but also a mechanism of data validation, where peers decide whether the common state is valid. This mechanism prevents one peer from suggesting a change that others may disagree with, and ensures that more than half of the peers accepted the change.

The proposed peer-set ledger structure is akin to a classic permissioned blockchain, however the global ledger is divided into multiple, separate ledgers maintained by a large number of peer-sets, which together participate in a global network.

C. Object and peer-set identification

One of the challenges in decentralized systems is object identification. A popular way of identifying objects in such

environments is *content addressing*. This method is usually based on attaching a cryptographic hash function’s digest of an object to its identifier. It addresses objects by their content instead of their location—it implies increasing data availability, but at the same time it restricts its modifications as every change generates a new identifier, and the old one becomes outdated. A well-known example is the BitTorrent protocol [17] which stores hashes inside torrent files or magnet links. Other uses include IPFS with CIDs [7], or Git with commit/tree/blob hashes [18].

When data is subject to change, content addressing becomes difficult to apply and other solutions are required. Standard identification methods, such as URLs, persistent identifiers, or handles [19], are content-agnostic, but in turn have fixed location, which undermines the decentralized nature of object storage and limits possibilities to migrate the data. Alternatives include *decentralized identifiers* (DIDs) [20], which are designed to identify and verify digital entities in decentralized web applications, and to provide a way of interacting with them. Blockchain also provides a possibility to store a public collection of peer-set identification records in a decentralized manner. However, all parties would have to agree to participate in a public network and to process the chain, which may discourage potential users.

Our architecture assumes both decentralized nature and possibility to modify data along with its owners. We propose a hybrid object identification system, which is based on a decentralized network of *peer-set identity management services* (PIMS), which are responsible for providing and managing information about peer-sets. An identifier to an object consists of three parts: 1) PIMS identification, 2) peer-set identification, and 3) object identification. The first part unambiguously identifies the service, which shares information about the peer-set upon request, using the second part of the identifier. This way, the user of the identifier may learn about the current state of the peer-set, without the need to change the identifier or waive the decentralized nature of the system (cf. f2). PIMSes are meant to be provided by organizations and create a decentralized network themselves.

Peer-set identity management services also have to implement specific identity management policies. Peer-sets are designed to be self-governing, so actions which modify a peer-set (such as adding a new peer, or removing an existing one) need to be established by the peer-set itself. Such requirement makes managing the data dependent on the data itself, which additionally justifies creating a dedicated service. For instance, adding a new peer D to a peer-set consisting of peers A , B , and C , may require approvals of D and majority of $\{A, B, C\}$.

D. Atomic commitment between peer-sets

Our proposal also takes into account atomic commitment between peer-sets. It may be used for instance to implement two-way relations between objects originating from different peer-sets. The atomic commitment ensures that two different peer-sets either both commit a change, or both ignore it (cf. entry 4 in table I and entry 6 in table II). We take into

consideration two ways of ensuring atomic commitment in our proposal.

Let us assume an atomic commitment between peer-sets A , B , and C , each one consisting of an arbitrary number of peers denoted A_0 , A_1 , etc. The first approach is to treat $\{A, B, C\}$ as one single database divided into three shards A , B , and C . Each shard contains a number of replicas, which are represented by peers. Using this interpretation we can use existing atomic commitment protocols which are designed for shards of replicas [21], but requires cooperation with the consensus protocol.

The second approach is to treat each peer-set as an entity and synchronize commitment between two representative peers. A special entry is added to the ledger, which represents an ongoing atomic commitment between peer-sets. This entry is used as a way of synchronizing state between all peers inside each of the peer-sets, and is used to implement a higher-level commitment protocol, which does not have to assume multiple replicas. Examples of such protocols include two-phase commit protocol (2PC), along with variations and improved versions such as 2PC* [22].

IV. FUTURE WORK

The problem of increasing data availability and fault tolerance in decentralized collaborative systems is very broad and complex. This publication aims to designate a vision of our solution, which is based on replication and decentralized authority over data, and outline directions of its development. We propose an architecture of a layer of data-sharing systems, consisting of

- peer-sets, which implement a decentralized mechanism of collaboration between peers,
- peer-set identity management services, which allow discovery and identification of peer-sets, and
- atomic commitment protocol, which enables performing atomic changes between peer-sets in order to implement e.g. two-way relations between objects.

Discussions in section III are entry points to more detailed research in each subject. We plan to evaluate several consensus protocols in terms of usability and integrate them with the idea of voting and validating changes by peers. In case of atomic commitment, we want to implement and compare the two proposed solutions. Finally, we plan to create a proof-of-concept of the whole system along with detailed description and evaluation. We also intend to integrate our solution with Onedata by providing an integration layer with GraphSync—the metadata synchronization protocol.

V. ACKNOWLEDGEMENTS

This scientific work was published in part by an international project co-financed by the program of the Minister of Science and Higher Education entitled “PMW” in the years 2021–2023; contract No. 5193/H2020/2021/22. KJ, RGS and JK are grateful for support from the subvention of Polish Ministry of Education and Science assigned to AGH University of Science and Technology.

REFERENCES

- [1] D. Cesini *et al.*, “The eXtreme-DataCloud project solutions for data management services in distributed e-infrastructures,” *EPJ Web of Conferences*, vol. 245, p. 04010, 01 2020. doi: 10.1051/epj-conf/202024504010
- [2] ScienceMesh. [Online]. Available: <https://sciencemesh.io/>
- [3] M. Wrzeszcz, Ł. Dutka, R. G. Słota, and J. Kitowski, “New approach to global data access in computational infrastructures,” *Future Generation Computer Systems*, vol. 125, pp. 575–589, 2021. doi: 10.1016/j.future.2021.06.054
- [4] L. Opióła, L. Dutka, R. G. Słota, and J. Kitowski, “Trust-driven, decentralized data access control for open network of autonomous data providers,” in *2018 16th Annual Conference on Privacy, Security and Trust (PST)*, 2018. doi: 10.1109/PST.2018.8514209 pp. 1–10.
- [5] Arora, Ishank, Alfageme Sainz, Samuel, Ferreira, Pedro, Gonzalez Labrador, Hugo, and Moscicki, Jakub, “Enabling interoperable data and application services in a federated sciencemesh,” *EPJ Web Conf.*, vol. 251, p. 02041, 2021. doi: 10.1051/epjconf/202125102041
- [6] T. T. A. Dinh, R. Liu, M. Zhang, G. Chen, B. C. Ooi, and J. Wang, “Untangling blockchain: A data processing view of blockchain systems,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 7, pp. 1366–1385, 2018. doi: 10.1109/TKDE.2017.2781227
- [7] J. Benet, “IPFS - content addressed, versioned, P2P file system,” *arXiv preprint arXiv:1407.3561*, 2014. doi: 10.48550/arXiv.1407.3561
- [8] D. Ulybyshev, M. Villarreal-Vasquez, B. Bhargava, G. Mani, S. Seaberg, P. Conoval, R. Pike, and J. Kobes, “(WIP) Blockhub: Blockchain-based software development system for untrusted environments,” in *2018 IEEE 11th International Conference on Cloud Computing (CLOUD)*, 2018. doi: 10.1109/CLOUD.2018.00081 pp. 582–585.
- [9] M. Pease, R. Shostak, and L. Lamport, “Reaching agreement in the presence of faults,” *J. ACM*, vol. 27, no. 2, p. 228–234, apr 1980. doi: 10.1145/322186.322188
- [10] L. Lamport, “Paxos made simple,” *ACM SIGACT News (Distributed Computing Column)* 32, 4 (Whole Number 121, December 2001), pp. 51–58, 2001.
- [11] I. Moraru, D. G. Andersen, and M. Kaminsky, “There is more consensus in egalitarian parliaments,” in *Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems Principles*, ser. SOSP '13. New York, NY, USA: Association for Computing Machinery, 2013. doi: 10.1145/2517349.2517350. ISBN 9781450323888 p. 358–372.
- [12] D. Ongaro and J. Ousterhout, “In search of an understandable consensus algorithm,” in *Proceedings of the 2014 USENIX Conference on USENIX Annual Technical Conference*, ser. USENIX ATC'14. USA: USENIX Association, 2014. ISBN 9781931971102 p. 305–320.
- [13] F. P. Junqueira, B. C. Reed, and M. Serafini, “Zab: High-performance broadcast for primary-backup systems,” in *2011 IEEE/IFIP 41st International Conference on Dependable Systems Networks (DSN)*, 2011. doi: 10.1109/DSN.2011.5958223 pp. 245–256.
- [14] P. R. Nair and D. R. Dorai, “Evaluation of performance and security of proof of work and proof of stake using blockchain,” in *2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, 2021. doi: 10.1109/ICICV50876.2021.9388487 pp. 279–283.
- [15] M. Castro and B. Liskov, “Practical byzantine fault tolerance,” in *Proceedings of the Third Symposium on Operating Systems Design and Implementation*, ser. OSDI '99. USA: USENIX Association, 1999. ISBN 1880446391 p. 173–186.
- [16] Amazon Quantum Ledger Database (QLDB). [Online]. Available: <https://aws.amazon.com/qldb/>
- [17] The BitTorrent protocol specification. [Online]. Available: https://www.bittorrent.org/beps/bep_0003.html
- [18] Git. [Online]. Available: <https://git-scm.com/>
- [19] Handle.Net Registry. [Online]. Available: <https://www.handle.net/>
- [20] World Wide Web Consortium. Decentralized identifiers (DIDs) v1.0. [Online]. Available: <https://www.w3.org/TR/did-core/>
- [21] S. Maiyya, F. Nawab, D. Agrawal, and A. E. Abbadi, “Unifying consensus and atomic commitment for effective cloud data management,” *Proc. VLDB Endow.*, vol. 12, no. 5, p. 611–623, jan 2019. doi: 10.14778/3303753.3303765
- [22] P. Fan, J. Liu, W. Yin, H. Wang, X. Chen, and H. Sun, “2PC*: A distributed transaction concurrency control protocol of multi-microservice based on cloud computing platform,” *J. Cloud Comput.*, vol. 9, no. 1, jul 2020. doi: 10.1186/s13677-020-00183-w

Network Systems and Applications

MODERN network systems encompass a wide range of solutions and technologies, including wireless and wired networks, network systems, services, and applications. This results in numerous active research areas oriented towards various technical, scientific and social aspects of network systems and applications. The primary objective of Network Systems and Applications conference track is to group network-related technical sessions and promote synergy between different fields of network-related research.

The rapid development of computer networks including wired and wireless networks observed today is very evolving, dynamic, and multidimensional. On the one hand, network technologies are used in virtually several areas that make human life easier and more comfortable. On the other hand, the rapid need for network deployment brings new challenges in network management and network design, which are reflected in hardware, software, services, and security-related problems. Every day, a new solution in the field of technology and applications of computer networks is released. The NSA track is devoted to emphasizing up-to-date topics in networking systems and technologies by covering problems and challenges related to the intensive multidimensional network developments. This track covers not only the technological side but also the societal and social impacts of network developments. The track is inclusive and spans a wide spectrum of networking-related topics.

The NSA track is a great place to exchange ideas, conduct discussions, introduce new ideas and integrate scientists, prac-

tioners, and scientific communities working in networking research themes.

TOPICS

- Networks architecture
- Networks management
- Quality-of-Service enhancement
- Performance modeling and analysis
- Fault-tolerant challenges and solutions
- 5G developments and applications
- Traffic identification and classification
- Switching and routing technologies
- Protocols design and implementation
- Wireless sensor networks
- Future Internet architectures
- Networked operating systems
- Industrial networks deployment
- Software-defined networks
- Self-organizing and self-healing networks
- Multimedia in Computer Networks
- Communication quality and reliability
- Emerging aspects of networking systems

Track 3 includes technical sessions:

- Complex Networks—Theory and Application (1st Workshop CN-TA'22)
- Internet of Things—Enablers, Challenges and Applications (6th Workshop IoT-ECAW'22)
- Cyber Security, Privacy and Trust (3rd International Forum NEMESIS'22)

An Integer Programming Approach Reinforced by a Message-passing Procedure for Detecting Dense Attributed Subgraphs

Arman Ferdowsi

Vienna University of Technology-ECS Group

K. N. Toosi University of Technology-Department of Mathematics

Email: aferdowsi@ecs.tuwien.ac.at

armanferdowsi@email.kntu.ac.ir

Abstract—One of the recent challenging but vital tasks in graph theory and network analysis, especially when dealing with graphs equipped with a set of nodal attributes, is to discover subgraphs consisting of highly interacting nodes with respect to the number of edges and the attributes' similarities. This paper proposes an approach based on integer programming modeling and the graph neural network message-passing manner for efficiently extracting these subgraphs. The experiments illustrate the proposed method's privilege over some alternative algorithms known so far, utilizing several well-known instances.

Index Terms—Graph partitioning, Network analysis, Integer programming, Message passing, Local search.

I. INTRODUCTION

WHEN studying graphs/networks, we usually face collections consisting of nodes with common topological and nodal attribute characteristics. More specifically, sets of highly interactive vertices are likely to yield and share common relationships and properties. In this sense, in all scientific fields where graphs are somehow implicated, one of the challenging tasks would be to efficiently partition the given graph into a number of dense subgraphs consisting of massively connected vertices that share similar properties. These subgraphs are well known as *communities*, and naturally, the process of identifying them is referred to as the *community detection problem*. Detecting communities has become one of the fundamental subjects in the field of network analysis and graph theory and has numerous applications in a wide range of areas, including the analysis of Social/Biological/Cosmological networks [1], [2], [3], [4] and WEB [5]. It also plays a crucial role in the domain of Network Design problems [6], Signal Processing [7], Image Segmentation [8], Pattern Recognition [9], and Data Mining [10].

Crucial in this domain is a more formal representation of a graph: A pair $G = (V, E)$ with the set of vertices V and edges E . Subsequently, from the perspective of graph topological structure, a *community* can be contemplated as a subset $C \subseteq V$ with a high density of edges between nodes inside C and a low density of edges connecting C to the other subsets. Accordingly, one can define the community detection problem as *partitioning* V into a set of disjoint communities $\mathbf{C} = \{C_1, C_2, \dots, C_k\}$.

In fact, mining high-quality communities usually coincides with finding a measure that estimates the goodness of communities. In the literature, a large number of such quality measures have been proposed for evaluating the superiority of partitioning from the viewpoint of topological structures. Among them, one of the most widely used and well-known is *Modularity*, introduced by Newman [11]. Intuitively, for a community C , Modularity is the number of edges inside C subtracted by the expected number of such edges, whereat the expected number of edges can be derived by corresponding to G , a randomized graph (called the *Null model*) with exactly the same vertices and the same degree of G , in which edges are placed randomly. More specifically if d_i and m are, respectively, the degree of node i and the number of edges in G , the probability of existing an edge between nodes i and j in such a graph is $\frac{d_i d_j}{2m}$. This is because, first, each node is assigned the number of stub links exactly equal to its edges (Fig. 1a, 1b). Afterward, each of the two stub edges will be joined at random (Fig. 1c). Consequently, the Modularity value for a community C can be defined as the number of edges within C in G minus the number of edges within C in a Null model of G . Thus, Modularity for partitioning \mathbf{C} can then be expressed as

$$Q(\mathbf{C}) = \frac{1}{2m} \sum_{i,j \in V} [a_{i,j} - \frac{d_i d_j}{2m}] \xi(i, j), \quad (1)$$

where $A = (a_{i,j})$ be the adjacency matrix of G , where $a_{i,j}$ is one when there is an edge between node i and node j , and zero otherwise; n is the number of vertices in G . In addition, $\xi(i, j)$ is one if i and j are in the same community and zero otherwise.

As a result, high-quality communities can be determined as the ones with a high value of Modularity. However, despite Modularity's advancements in finding high-quality communities in a wide range of graphs, it is known to suffer from limitations (see [12], [13], for example). In particular, as pointed out in [14], since Modularity only considers the existing edges of the network, it qualifies the goodness of the discovered communities by only measuring how good the partitioning fits the existing edges. This is indeed a drawback

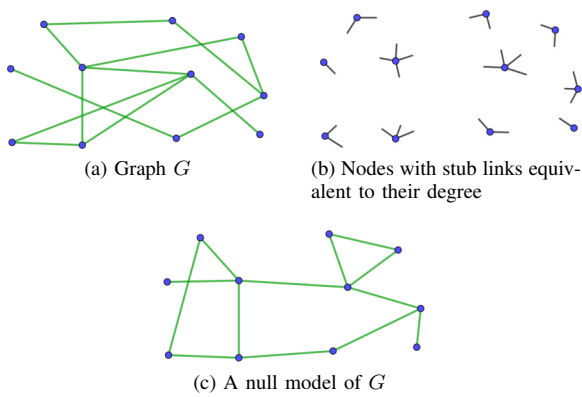


Fig. 1: A Null model associated with a given Graph G .

because the disconnected nodes (absent links) that lie in the same community are not taken into account.

On the other hand, *Max-Min Modularity* [14], as one of the successful extensions of Modularity, is able to significantly improve the accuracy of the measure by compensating for the Modularity quantity when disconnected nodes are in the same community. More precisely, it is assumed in [14] that (in addition to the graph G) a zero-one *relation matrix* $U = (u_{i,j})$ is given that defines whether every pair of disconnected nodes of the network is related or not: $u_{i,j}$ is one when disconnected nodes i and j are *related*, and zero otherwise. Max-Min Modularity, in fact, tries to take into account the importance of the *indirect* connections between disconnected nodes by penalizing the Modularity measure when *unrelated* nodes are in the same community: Consider a complemented graph $G' = (V, E')$, where E' contains an edge between every pair of disconnected nodes of G that is unrelated; i.e., there is an edge between i and j in G' if there is not such an edge in G and also $u_{i,j}$ is zero. Max-Min Modularity, then, aims at maximizing the Modularity in G as well as minimizing Modularity in G' simultaneously. Mathematically speaking, if $A' = (a'_{i,j})$ is the adjacency matrix of G' , d'_i is the degree of node i in G' , and m' is the number of the edges in G' , then Max-Min Modularity Q_{MM} of a given partition \mathcal{C} of V is defined as follows:

$$Q_{MM}(\mathcal{C}) = \sum_{i,j \in V} \left[\frac{1}{2m} (a_{i,j} - \frac{d_i d_j}{2m}) - \frac{1}{2m'} (a'_{i,j} - \frac{d'_i d'_j}{2m'}) \right] \xi(i, j). \quad (2)$$

We refer to the problem of finding a partition of the network that maximizes Max-Min Modularity as the *Max-Min Modularity Maximization* problem.

As a crucial remark, we can note that not only is the Max-Min Modularity Maximization problem categorized in the class of NP-hard problems [15], making it hard to find an efficient optimization algorithm, but it also suffers from a major drawback: Max-Min Modularity strongly depends on the accuracy of the given relation matrix, meaning that the quantity of the measure might be heavily affected by the node relationships defined by the user/oracle in the first place. De-

spite Ferdowsi and Khanteymooari [16] succeeded in proposing a systematic approach for unveiling the relation between non-adjacent vertices, from a more critical point of view, one could argue that both Modularity and Max-Min Modularity, like many other community discovery methods, only utilize topological information of nodes, naively overlooking a rich set of nodal attributes (e.g., user profiles of an online social network or textual contents of a citation network), which is abundant in all real-life networks [17].

In this regard, recently, an emerging domain of deep learning for graphs has ensured the design of more accurate and scalable algorithms. However, despite these approaches achieving outstanding results in graph-related tasks like link prediction and node classification [18], fairly little concentration has been committed to their application on unsupervised learning that encompasses the community detection problem. This is primarily because graph embedding methods can ideally align with (semi-)supervised learning approaches; however, they cannot be naturally generalized to the unsupervised learning manners since it is not very simple to find a proper loss function to govern the back-propagation updating procedure. Despite that, several deep learning-based unsupervised graph algorithms have been proposed [19], [20], [21], but they all suffer from a not accurate choice of a loss function that can authoritatively extend the model to unsupervised vision. The diverse results obtained by these approaches over the same input instances can prove this claim.

Main contribution: In this work, we introduce a refined model for the Max-Min Modularity that empowers us to find high-quality communities with simultaneously taking the graph's topological structure and the nodal attributes into account. In the sequel, we involve the Modularity metric to mine the communities consisting of densely connected vertices. In addition, we provide a technical introduction to *Graph Neural Network (GNN)* formalism, a dominant and fast-growing paradigm for deep learning with graph data, whose ability is to use nodes' features and local structure to generate embeddings. Utilizing the general concept of GNN feed-forward message passing, we devise an efficient mechanism for extracting the information induced by propagating nodes' properties throughout the network, leading to a perfect systematic characterization of the relation matrix. Everything combined, we introduce the advanced Max-Min Modularity scheme and express it with a standard integer programming formulation. Ultimately, by solving the model using a robust approach consisting of a row/column generation technique for solving the model's linear relaxation version and a local search manner for obtaining integer solutions, we identify the final communities.

The paper is organized as follows: the rest of this section focuses on providing a brief literature review. In Section II, we first restate the Modularity Maximization problem in terms of an integer programming model. We then devise a method for providing a refined relation matrix. Afterward, gathering all things together, we extend the primary Max-Min Modularity model and present an integer programming formulation for

it. Next, in Section III, we introduce the employed solution approach that leads to discovering high-quality communities. Section IV eventually focuses on various experiments, confirming the proposed method's high performance.

A. Related Works

Several approaches have been proposed in the literature to detect communities in networks; see, for example, the survey conducted by Souravlas et al. [22]. However, despite a large number of these techniques, relatively little work solves the problem using mathematical programming techniques. Especially in the case of Modularity maximization, few results have been established [23], [24], [25], [26]. On the other side, Chen et al. [14] illustrated that maximizing the Modularity alone does not usually lead to superior communities. Based on their findings, one of the significant drawbacks of Modularity is its sheer dependence on the (existing) links, meaning that, Modularity only focuses on discovering communities in which the number of interactive edges (links) is as many as possible. At the same time, it does not pay any attention to the missing links. This is while just as existing links can play an essential role in analyzing networks, so do the absent links. Therefore, new extensions of Modularity emerged subsequently, one of the most successful of which is the already mentioned: Max-Min Modularity [14]. Nevertheless, as discussed, Max-Min Modularity itself suffers from a critical issue: the nonexistence of a systematic way for proposing the so-called relation matrix, which is required to express the relationship between non-adjacent nodes. In this respect, Ferdowsi and Khanteymooori [16] succeeded in offering an analytical procedure to address this deficiency, generalize the conventional Max-Min Modularity, and provide an efficient local search-based algorithm to discover high-quality communities by considering both existing and missing links.

On the other side, in the past few years, most research has surged toward deep learning methods due to their power to achieve unprecedented results. These techniques aim at embedding nodes into a low-dimensional, dense vector space [27], [28]. However, unfortunately, a vast number of these methods lack the strength of encountering attributed graphs in which nodes are equipped with a set of features, and this is while we are now most surrounded by attributed networks everywhere [19]. It is worth mentioning that a few deep-learning methods have been recently proposed that consider attributed network embedding [29], [30], though most of them employ a matrix factorization manner, which endures some critical boundaries. More precisely, the representation capability of a matrix factorization-based approach is found to be more inadequate than a neural network-based method [31]. Besides, one could also argue that the majority of these proposals only rely on supervised graph algorithms, and therefore, usually fail to perform the community detection task, which can be categorized as an unsupervised assignment in graph problems [32], [33]. And the rests, which are designed for unsupervised learnings, still cannot find a promising loss function that can

lead to fine communities for various given graph instances [34], [19].

II. MODEL SKETCHING

In this section, we first recite the so-called Modularity Maximization problem in terms of an integer programming formulation, enabling us to discover communities with respect to the topological aspects of a given graph. Afterward, we explain the Graph Neural Network message passing method that facilitates us to devise a procedure for determining an accurate relation matrix for the Max-Min Modularity problem. This achievement then hopefully leads to a proper integer formulation for our advanced Max-Min Modularity problem that can be used to efficiently capture communities with respect to simultaneously taking both topological and attributes aspects into consideration.

A. Topology extraction

Let the binary variable x_{ij} indicate if nodes i and j belong to the same community or not; the value of x_{ij} is zero if nodes i and j belong to the same community, and one otherwise. Let $I_{all} = \{(i, j) \in V^2 \mid i < j\}$; and $q_{ij} = a_{i,j} - \frac{d_i d_j}{2m}$, for each $(i, j) \in I_{all}$. As described in [25], the Modularity Maximization problem can be formulated in terms of the following integer linear program:

$$\max \frac{1}{m} \sum_{(i,j) \in I_{all}} q_{ij}(1 - x_{ij}) \quad (\text{IP-M})$$

$$x_{ij} + x_{jk} - x_{ik} \geq 0 \quad \forall i < j < k \quad (3)$$

$$x_{ij} - x_{jk} + x_{ik} \geq 0 \quad \forall i < j < k \quad (4)$$

$$-x_{ij} + x_{jk} + x_{ik} \geq 0 \quad \forall i < j < k \quad (5)$$

$$x_{ij} \in \{0, 1\} \quad \forall (i, j) \in I_{all} \quad (6)$$

Constraints (3)-(5) guarantee that if i and j are in the same community and j and k are in the same community, then so are i and k . We refer to the relaxation of (IP-M), obtained by replacing the constraints $x_{ij} \in \{0, 1\}$ by $x_{ij} \in [0, 1]$, as (LP-M).

As discussed in [23] and [35], solving IP-M can soundly provide us with communities consisting of highly interactive edges. It is, however, incontrovertible that maximizing Modularity cannot help us tackle the attributed graphs. Consequently, to address this deficiency, our goal is to design an advanced model for the Max-Min Modularity problem so that nodal attributes are also effectively involved in the community mining process. In order to do that, we first provide a way to exploit the information diffused via nodes' features.

B. Attribute extraction

In this section, which establishes the core part of this research, we utilize the *Graph Neural Network (GNN) message-passing* framework to provide a systematic and accurate way of defining a *relation matrix* that perfectly depicts the similarity between nodes by analyzing the information spread by the attributes. More precisely, we aim to feed the nodal attributes to a GNN message passing to create single representation

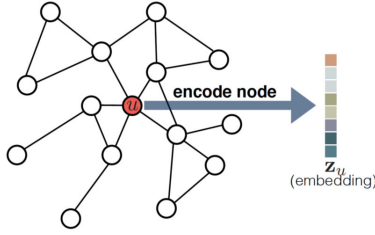


Fig. 2: A scheme of the encoder approach. The encoder maps the node u to a low-dimensional embedding z_u .

vectors, each of which captures a node's structure as well as the feature information.

To motivate our discussion, we initially raise the notion of *node embedding*, which seeks to encode nodes as low-dimensional vectors that summarize their structural graph position and the information of their local graph neighborhood. In other words, node embedding aims to project vertices into a latent space, where geometric relations in this latent space correspond to relationships (e.g., edge existence or similarity) in the original graph or network. In this fashion, an encoder can be referred to as a function that maps each node $u \in V$ to vector embedding $z_u \in \mathbb{R}^d$ (i.e., z_u corresponds to the embedding for node $u \in V$) (See Fig. 2).

We now turn our attention toward one of the substantial encoder models: *Graph Neural Network (GNN)*, a broad framework for defining deep neural networks on graph data. The elucidative characteristic of a GNN is that it adopts a form of neural message passing in which vector messages are exchanged between nodes and updated using neural networks [36]. In each message-passing iteration of a GNN, a hidden embedding $h_u^{(k)}$ corresponding to each node $u \in V$ is updated according to information aggregated from u 's graph neighborhood $\mathcal{N}(u)$. Informally speaking, to each node $u \in V$ one can correspond a so-called *computational graph* that accumulates the information propagated from u 's neighbors, and in turn, the messages coming from these neighbors are based on information aggregated from their respective neighborhoods, and so on. Fig. 3 exemplifies a two layers computational graph. From the mathematical perspective, GNN message-passing procedure can be expressed as follows. Suppose that each node u is associated with a d -dimensional attribute vector that we represent with $X_u \in \mathbb{R}^d$. Then, the k -th embedding layer $h_u^{(k)}$, corresponding to node $u \in V$, can be obtained by the following recursive formula:

$$h_u^{(k)} = \sigma(W^{(k)} \sum_{v \in \mathcal{N}(u)} h_v^{(k-1)} + b^{(k)}), \quad (7)$$

where $h_u^{(0)} = X_u$ and $W_{d \times d}$ is a trainable matrix, which weights the nodes' attributes. Moreover, let σ denotes an elementwise non-linearity (e.g., a *tanh* or *Relu*). Furthermore, $b^{(k)} \in \mathbb{R}^d$ is the bias term, which can be often omitted for the sake of simplicity, but including it could be important to obtain high-quality performance. As can be inferred from (7), first, the messages incoming from the neighbors are summed;

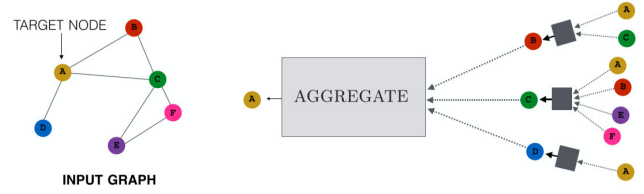


Fig. 3: Two layers computational graph corresponding to node A . The model aggregates messages from A 's local graph neighbors, and in turn, the messages coming from these neighbors are based on information aggregated from their respective neighborhoods.

then, the neighborhood information with the node's previous embedding is combined using a linear combination; finally, an elementwise non-linearity is applied.

Now, in order to transform the node-level equation (7) into something we can implement, we can come up with the following graph-level definition of the model:

$$H^{(k)} = \sigma(AH^{(k-1)}W^{(k)}), \quad (8)$$

where $H^{(k)} \in \mathbb{R}^{|V| \times d}$ is the matrix of node representations at layer k in the GNN, with each node corresponding to a row in the matrix.

However, despite the straightforward intuition behind the update procedure (8), considerably notable is its two limitations. The first one is the non-consideration of the attributes of each node itself. This is crucial since the effectiveness of GNN becomes severely limited, as the information coming from the node's neighbors cannot be differentiated from the information from the node itself. This restriction originates from the fact that self-loops are not taken into account in the adjacency matrix A . The problem, however, can be easily resolved by substituting A with $A + I$, where $I \in \mathbb{R}^{n \times n}$ is the identity matrix that lets the self-loops also be involved. Another issue that may raise concern regarding (8) is the non-normality of features, which can indeed lead to numerical instabilities. However, fortunately, to also prevent this shortcoming, it seems convincing to apply the symmetric normalization as it has turned out to drive powerful dynamics [37]. For doing this, it is sufficient to replace $A + I$ with the normalization term $D^{-\frac{1}{2}}(A + I)D^{\frac{1}{2}}$, where $D \in \mathbb{R}^{n \times n}$ is the degree matrix. As a result, in the case of a basic GNN, we can end up with the following graph level definition of the model:

$$H^{(k)} = \sigma(D^{-\frac{1}{2}}(A + I)D^{\frac{1}{2}}H^{(k-1)}W^{(k)}) \quad (9)$$

It is worth pointing out that since our goal is to employ the feed-forward message passing exclusively and not to implement the back-propagation procedure, training the weight matrix W is not specifically part of our requirements. Instead, the only thing that matters to us is finding a final vector representation for each node by which we can sufficiently reflect the similarities among nodes with respect to the information that originates/propagates from attributes. For this reason, we can

safely omit W from (9) and obtain the following embedding matrix H .

$$H^{(k)} = \sigma(D^{-\frac{1}{2}}(A + I)D^{\frac{1}{2}}H^{(k-1)}) \quad (10)$$

One important insight that could be gained by (10) is that GNN feed-forward message passing is capable of effectively encoding neighborhood information in such a way that after performing the updating procedure for a number of layers, similar nodes in the graph will tend to have analogous final embedding representation [38]. This is primarily due to the fact that each node inherits the information (attributes) from its neighborhood.

Accordingly, any techniques for computing the distance between each pair of final vectors representation could naturally lead to obtaining the similarities between the corresponding vertices with respect to their attributes. In this work, we employ the well-known *z-Normalized Euclidean Distance* to measure the similarities between nodes. In this fashion, the distance between two vectors is defined as the Euclidean distance between the normal form of the two vectors, where the normalized form associated with each sequence is obtained by transforming the vector so it has a mean distribution $\mu = 0$ and standard deviation $\sigma = 1$. More explicitly, given two nodes i and j , with the corresponding vector embeddings $H_i^{(k)} \in \mathbb{R}^d$ and $H_j^{(k)} \in \mathbb{R}^d$, at layer k , we define $x_{ij}^* = \frac{\|H_i^{(k)} - H_j^{(k)}\|_2}{2\sqrt{d}}$ to be their z -score normalized Euclidean distance, where $\|\cdot\|_2$ is the Euclidean norm, $\widehat{H_i^{(k)}} = \frac{H_i^{(k)} - \mu_{H_i^{(k)}}}{\sigma_{H_i^{(k)}}}$, $\mu_{H_i^{(k)}}$ is the distribution mean, and $\sigma_{H_i^{(k)}}$ is the standard deviation of the vector $H_i^{(k)}$. It is not difficult to check that for any given $i, j \in V$, we have $x_{ij}^* \in [0, 1]$ [39] and that x^* preserves the triangle inequality. Subsequently, x^* forms a metric space, which we denote by *Embedding Distance (ED)*, on graph G . Clearly, the larger the obtained ED between two nodes, the less similarity between them.

C. Advanced Max-Min Modularity Model

As already mentioned, the larger x_{ij}^* is, the less likely it is that i and j are similar, that is, the less correlated they are, and therefore, the more likely they are to be in distinct communities. This intuition and also the fact that the Modularity Maximization problem can be nicely expressed for weighted graphs [40] persuade us to propose an advanced Max-Min Modularity model by recharacterizing the relation matrix and so the complemented weighted graph $G' = (V, E')$ using ED. Accordingly, we define the relation matrix $A' = (a'_{i,j})$ and G' ($(a'_{i,j})$ represents the weight of the edge between nodes i and j in G') as follows:

$$a'_{i,j} = \begin{cases} x_{ij}^* & \text{if } a_{i,j} = 0 \text{ and } i \neq j \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

Given a relation matrix $A' = (a'_{i,j})$, and noticing that in the induced metric ED, Constraints (3)-(5) guarantee the

triangle inequality, for any $i, j, k \in V$, the advanced Max-Min Modularity Maximization problem can be formulated as the following IP:

$$\begin{aligned} \max \quad & \sum_{(i,j) \in I_{all}} \left(\frac{q_{ij}}{m} - \frac{q'_{ij}}{m'} \right) (1 - x_{ij}) \quad (\text{IP-MM}) \\ & (3), (4), (5) \\ & x_{ij} \in \{0, 1\} \quad \forall (i, j) \in I_{all}, \end{aligned}$$

where $q'_{ij} = a'_{i,j} - \frac{d'_i d'_j}{2m'}$, for each $(i, j) \in I_{all}$; $d'_i = \sum_{j=1}^n a'_{i,j}$, and $m' = \sum_{(i,j) \in I_{all}} a'_{i,j}$. We refer to the relaxation of IP-MM, obtained by replacing the constraints $x_{ij} \in \{0, 1\}$ by $x_{ij} \in [0, 1]$, as (LP-MM).

Considerably important is the fact that maximizing IP-MM coincides with simultaneously maximizing the modularity over the original given graph G and minimizing the modularity over the complemented graph G' , determined by the proposed message-passing approach. Whereas the first component makes sure to return communities, each consisting of densely connected nodes, the latter attempts to extract the communities containing vertices with the most similar attributes possible.

III. SOLUTION APPROACH

Efficiently solving (IP-MM) consists of two main parts: 1) optimally solving (LP-MM) and 2) accurately rounding the obtained fractional solutions to the integer ones.

For the first task, we use the row/column generation algorithm proposed in [16] that perfectly works for (LP-MM) as well since the two models are identical apart from using different relation matrices. A summary of the applied row/column generation technique is as follows. First, consider the following sub-problem (LPs-MM(\mathcal{I})) of (LP-MM) consisting of all pairs in $I = \{(i, j) \in I_{all} \mid c_{ij} > 0\}$ and some pairs in $I' = \{(i, j) \in I_{all} \mid c_{ij} \leq 0\}$:

$$\begin{aligned} \max \quad & \sum_{(i,j) \in I} c_{ij}(1 - x_{ij}) + \sum_{(i,j) \in \mathcal{I} \cap I'} c_{ij}(1 - x_{ij}) \quad (\text{LPs-MM}(\mathcal{I})) \\ & x_{ij} + x_{jk} - x_{ik} \geq 0 \quad \forall (i, j), (j, k), (i, k) \in I \cup \mathcal{I}, c_{ij} \geq 0 \vee c_{jk} \geq 0 \quad (12) \\ & x_{ij} - x_{jk} + x_{ik} \geq 0 \quad \forall (i, j), (j, k), (i, k) \in I \cup \mathcal{I}, c_{ij} \geq 0 \vee c_{ik} \geq 0 \quad (13) \\ & -x_{ij} + x_{jk} + x_{ik} \geq 0 \quad \forall (i, j), (j, k), (i, k) \in I \cup \mathcal{I}, c_{jk} \geq 0 \vee c_{ik} \geq 0 \quad (14) \\ & x_{ij} \in [0, 1] \quad \forall i < j, (i, j), (j, k), (i, k) \in I \cup \mathcal{I} \quad (15) \end{aligned}$$

Be advised that (LPs-MM(\emptyset)) generates the smallest formulation while (LPs-MM(I')) is equivalent to (LP-MM) itself. Moreover, point out that (LPs-MM(\mathcal{I})) delivers an upper bound of the optimal value of (LP-MM) that never gets worse by adding variables [16]. Above all, however, as Theorem 3.1 in [16] depicts, if an optimal solution $\bar{x}^* = (x_{ij}^*)_{(i,j) \in I \cup \mathcal{I}}$ to (LPs-MM(\mathcal{I})) satisfies the following condition (*), then $(x_{ij}^*)_{(i,j) \in I_{all}}$ is an optimal solution to (LP-MM), where

$$x_{ij}^* = \begin{cases} \bar{x}_{ij}^* & ; (i, j) \in I \cup \mathcal{I} \\ 1 & ; \text{otherwise} \end{cases} \quad (16)$$

and

$$(*) \begin{cases} x_{ij}^* + x_{jk}^* \geq 1; (i, j), (j, k) \in I \cup \mathcal{I}, c_{ij} \geq 0 \vee c_{jk} \geq 0, (i, k) \in I' - \mathcal{I} \\ x_{ij}^* + x_{ik}^* \geq 1; (i, j), (i, k) \in I \cup \mathcal{I}, c_{ij} \geq 0 \vee c_{ik} \geq 0, (j, k) \in I' - \mathcal{I} \\ x_{jk}^* + x_{ik}^* \geq 1; (j, k), (i, k) \in I \cup \mathcal{I}, c_{jk} \geq 0 \vee c_{ik} \geq 0, (i, j) \in I' - \mathcal{I} \end{cases}$$

This fact leads to the following efficient row/column generation procedure for optimally solving (LP-MM):

- Start solving (LPs-MM(\mathcal{I})) with $\mathcal{I} = \emptyset$ and adding those $x^* \in I' - \mathcal{I}$ that violate inequalities in (*) in each iteration, until an optimal solution to (LPs-MM(\mathcal{I})) satisfies (*).
- In each repeat, employ a row generation technique for solving (LPs-MM(\mathcal{I})):
 - 1) Obtain an optimal solution \bar{x}^* by solving (LPs-MM(\mathcal{I})) with no constraints.
 - 2) verify whether all constraints of (LPs-MM(\mathcal{I})) are satisfied by \bar{x}^* . If not, add the violated ones and solve (LPs-MM(\mathcal{I})).
 - 3) Update \bar{x}^* and repeat step (2).
- By having the optimal solution \bar{x}^* to (LPs-MM(\mathcal{I})), determine the optimal solution x^* to (LP-MM) by Equation (16).

To accomplish the second task, we can again apply the rounding algorithm proposed in [16], which can be shortly expressed as follows. Point out that the fractional optimal solution to (LP-MM) provides us with a metric space, where the distance between every pair of nodes is at most one. Let us call this metric *LP distance*. It is apparent that in an integral solution, the distance between each pair of nodes is either zero (implying that these two nodes belong to the same community) or one (inferring that these two nodes belong to separate communities). The rounding task is to push (some of) the nodes so as to determine a final valid *configuration* of the points in which the distance between every pair of nodes is either zero or one. Obviously, such a valid configuration correlates with a feasible integral solution \hat{x} to (IP-MM): $\hat{x}_{ij} = 0$, for points i and j which are co-located; and $\hat{x}_{ij} = 1$, for those with the distance one from each other. The main idea of the local search-based rounding procedure is to explore such promising configurations (using the LP information) and find a configuration leading to a solution (partitioning) whose max-min modularity value (2) is (locally) optimal.

Assume that \bar{x} is the optimal fractional solution to (LP-MM), obtained by the mentioned row/column generation technique. As explained above, to compute an integral solution to (IP-MM), we need to determine a set of final locations at distance one from each other (we refer to these final locations as community centers) and move the nodes to these centers so as to obtain a valid configuration. In fact, to compute an integral solution, we only need to determine a set of distinct centers, since we can then simply move each point to the closest center (with respect to the LP distance) and obtain a corresponding integral solution: for each pair i and j , $\hat{x}_{ij} = 0$ if i and j are co-located; and 1 otherwise. Therefore, the only task of the utilized local search algorithm is to determine a good set of final centers.

More precisely, the local search algorithm works as follows: Randomly pick a subset of V as initial centers. As discussed above, move other vertices to these centers to form a solution (partitioning) and then compute the max-min modularity value

TABLE I: Networks under study

ID	Network
1	CiteSeer [41]
2	Arnetminer [42]
3	Caltech36 [43]
4	Reed98 [43]
5	Facebook348 [38]

of the resulting partitioning; see (2). The local search technique tries to improve the max-min modularity value by adding and/or deleting a center to/from the set of centers at a time. The local search movement that yields the most significant improvement in the max-min modularity value is selected at each iteration. The algorithm terminates when no improving local search move exists.

IV. COMPUTATIONAL RESULTS

This section presents a comprehensive performance evaluation for the proposed method using five well-known real-world networks listed in Table I. Ground truth (i.e., the optimal community structures) is available and known for each of these networks, and therefore, one can easily measure the quality of a community detection algorithm by estimating the similarities between the communities obtained by the algorithm and the ground truth. For doing this, we use the well-known performance metrics *Adjusted Rand Index* and *Normalized Mutual Information*, explained in the following subsection.

A. Performance metrics

Suppose that for a given graph G , $\mathcal{C}(\mathcal{A}) = \{C_1, \dots, C_k\}$ and $\mathcal{C}' = \{C'_1, \dots, C'_{k'}\}$ be respectively a set of communities obtained by an algorithm \mathcal{A} and the ground truth.

Although NMI [44] is a well-known clustering comparison metric, it can perfectly evaluate the similarity between the optimal communities and those discovered by an algorithm. The NMI value corresponding to the algorithm \mathcal{A} can be written as

$$NMI = \frac{-2 \sum_{x=1}^{|\mathcal{C}|} \sum_{y=1}^{|\mathcal{C}'|} \frac{|C_x \cap C'_y|}{n} \log\left(\frac{n|C_x \cap C'_y|}{|C_x||C'_y|}\right)}{\sum_{x=1}^{|\mathcal{C}|} \frac{C_x}{n} \log\left(\frac{C_x}{n}\right) + \sum_{y=1}^{|\mathcal{C}'|} \frac{C'_y}{n} \log\left(\frac{C'_y}{n}\right)} \quad (17)$$

In the case where the detected communities are identical to the ground truth, the NMI takes its maximum value one, while in the case where the two sets totally disagree, the NMI score is zero. Generally, the more the NMI, the better community structures have been found.

Adjusted rand index (ARI) [45], associated with algorithm \mathcal{A} , measures the similarity between $\mathcal{C}(\mathcal{A})$ and \mathcal{C}' as follows

$$ARI = \frac{2(a \times d - b \times c)}{(a + b) \times (b + d) \times (a + c) \times (c + d)} \quad (18)$$

where a , b , c and d are respectively the number of vertex pairs that are in the same community in both $\mathcal{C}(\mathcal{A})$ and \mathcal{C}' , in the

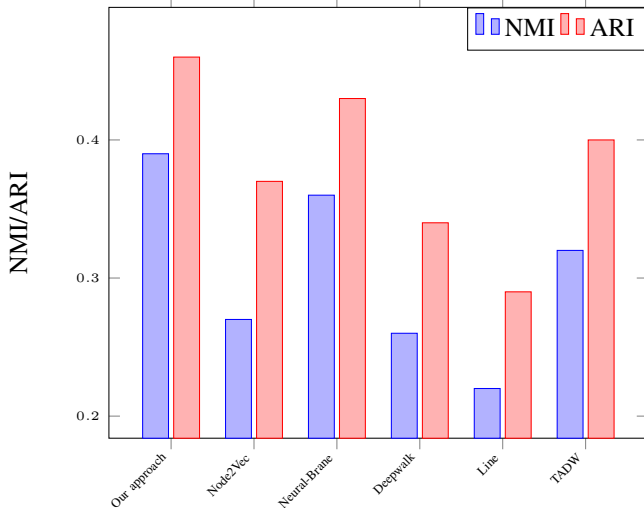


Fig. 4: Average normalized ARI and NMI performance rank of different algorithms applying to the real-world networks, presented in Table I. The average NMI and ARI values obtained by our method are respectively equal to 0.39 and 0.46.

same community in $\mathcal{C}(\mathcal{A})$ but not in \mathcal{C}' , in the same community in \mathcal{C}' but not in $\mathcal{C}(\mathcal{A})$, and in different communities in both $\mathcal{C}(\mathcal{A})$ and \mathcal{C}' . Like the NMI measure, the value of *ARI* varies between 0 and 1, and the higher its value is, the more similarity is between the communities obtained by algorithm \mathcal{A} and the grand truth of G .

B. Experiments

In what follows, we provide a process for comparing the performance of our proposed algorithm against five powerful rival algorithms: *Node2Vec* [28], *Neural-Brane* [19], *DeepWalk* [27], *Line* [46], and *Text-Associated DeepWalk (TADW)* [30]. These are all state-of-the-arts for integrating both network topology and nodal attributes for graph representation learning.

All algorithms are implemented with C++, and CPLEX optimizer 12.9 is used for solving linear programming.

Fig. 4 provides a comprehensive comparison by evaluating communities in terms of the average ARI and NMI ranks over all datasets. It is apparent that the proposed method significantly outperforms other algorithms in the sense that the communities it discovered are much more similar to the ground truths than those obtained by the other methods.

Although the results obtained in Fig. 4 perfectly illustrate the proposed method’s superiority and reliability against some other state-of-the-art algorithms, it could still be worth investigating the role of the applied GNN feed-forwards message passing procedure in improving the communities. For doing this, we compute the value of NMI associated with each of the networks’ obtained final communities in terms of different hidden layer numbers k . In this regard, $k = 0$ refers to the case when the embedding layer is not applied, and only Modularity is employed to identify communities with respect to the graph’s topological structure. Fig. 5 shows the results.

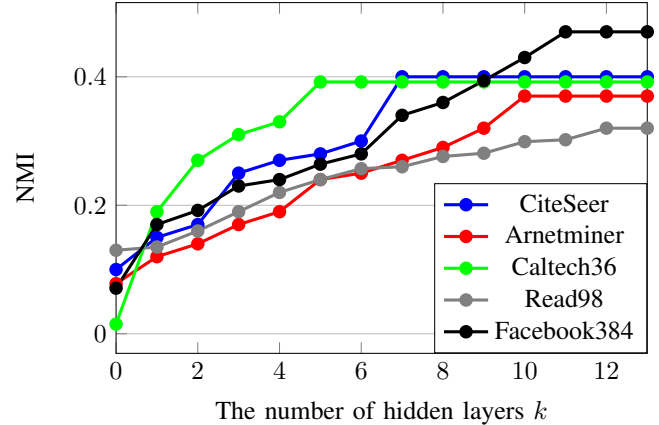


Fig. 5: The computed NMI corresponding to the final communities of each of the networks, provided in Table I, with respect to different values of k .

It is apparent from the results that, first of all, maximizing the Modularity alone (i.e., only relying on topological aspects and ignoring the nodal attributes) cannot lead to promising results. Furthermore, considerably notable is the approving effect of increasing the number of hidden layers. The more the value of k , the more accurate the embedding vector representation becomes due to the more information diffused through the neighboring vertices. However, interestingly enough, from a certain point on, increasing k does not influence the quality of communities, and this is primarily due to the fact that after a certain number of updates, each of the embedding vectors starts to converge.

V. CONCLUSION

In this work, we built a community discovery method on the basis of the mathematical programming formalism and the graph neural network feed-forward message passing manner. We managed to propose a systematic way to generate an authentic relation matrix for the Max-Min Modularity problem centered on an efficient node embedding technique, which enabled us to model the standard integer formulation for the Max-Min Modularity Maximization problem. A successful row/column generation technique and a local search-based rounding algorithm facilitated us in solving the model accurately and capturing the communities of a given attributed network. Furthermore, the proposed computational experiments showed that our results highly resemble the optimal solutions and that our algorithm outperforms the previous well-known algorithms.

REFERENCES

- [1] L. Jiang, L. Shi, L. Liu, J. Yao, and M. A. Yousuf, “User interest community detection on social media using collaborative filtering,” *Wireless Networks*, pp. 1–7, 2019.
- [2] A. Ferdowsi, M. Dehghan Chenary, and A. Khanteymooi, “Tscda: a dynamic two-stage community discovery approach,” *Social Network Analysis and Mining*, vol. 12, no. 1, pp. 1–14, 2022.

- [3] Y. Atay, I. Koc, I. Babaoglu, and H. Kodaz, "Community detection from biological and social networks: A comparative analysis of metaheuristic algorithms," *Applied Soft Computing*, vol. 50, pp. 194–211, 2017.
- [4] D. Krioukov, M. Kitsak, R. S. Sinkovits, D. Rideout, D. Meyer, and M. Boguñá, "Network cosmology," *Scientific reports*, vol. 2, p. 793, 2012.
- [5] S. Aparicio, J. Villazón-Terrazas, and G. Álvarez, "A model for scale-free networks: application to twitter," *Entropy*, vol. 17, no. 8, pp. 5848–5867, 2015.
- [6] A. Ferdowsi, M. DehghanChenari, F. Jolai, and R. Tavakkoli-Moghaddam, "Toward unraveling multi-objective optimization problems: A hybrid approach for solving a novel facility location problem." [7] N. Tremblay and P. Borgnat, "Graph wavelets for multiscale community mining," *IEEE Transactions on Signal Processing*, vol. 62, no. 20, pp. 5227–5239, 2014.
- [8] O. A. Linares, G. M. Botelho, F. A. Rodrigues, and J. B. Neto, "Segmentation of large images based on super-pixels and community detection in graphs," *IET Image Processing*, vol. 11, no. 12, pp. 1219–1228, 2017.
- [9] L. M. Freitas and M. G. Carneiro, "Community detection to invariant pattern clustering in images," in *2019 8th Brazilian Conference on Intelligent Systems (BRACIS)*. IEEE, 2019, pp. 610–615.
- [10] F. D. Malliaros and M. Vazirgiannis, "Clustering and community detection in directed networks: A survey," *Physics Reports*, vol. 533, no. 4, pp. 95–142, 2013.
- [11] M. E. Newman, "Modularity and community structure in networks," *Proceedings of the national academy of sciences*, vol. 103, no. 23, pp. 8577–8582, 2006.
- [12] S. Fortunato and M. Barthelemy, "Resolution limit in community detection," *Proceedings of the national academy of sciences*, vol. 104, no. 1, pp. 36–41, 2007.
- [13] T. Chakraborty, A. Dalmia, A. Mukherjee, and N. Ganguly, "Metrics for community analysis: A survey," *ACM Computing Surveys (CSUR)*, vol. 50, no. 4, pp. 1–37, 2017.
- [14] J. Chen, O. R. Zaiñane, and R. Goebel, "Detecting communities in social networks using max-min modularity," in *Proceedings of the 2009 SIAM international conference on data mining*. SIAM, 2009, pp. 978–989.
- [15] U. Brandes, D. Delling, M. Gaertler, R. Gorke, M. Hofer, Z. Nikoloski, and D. Wagner, "On modularity clustering," *IEEE transactions on knowledge and data engineering*, vol. 20, no. 2, pp. 172–188, 2007.
- [16] A. Ferdowsi and A. Khanteymooi, "Discovering communities in networks: A linear programming approach using max-min modularity," in *2021 16th Conference on Computer Science and Intelligence Systems (FedCSIS)*. IEEE, 2021, pp. 329–335.
- [17] A. R. Barghi, A. Ferdowsi, and A. Abhari, "Musical preferences prediction by classification algorithm," in *Proceedings of the Communications and Networking Symposium*, 2018, pp. 1–12.
- [18] H. Cai, V. W. Zheng, and K. C.-C. Chang, "A comprehensive survey of graph embedding: Problems, techniques, and applications," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 9, pp. 1616–1637, 2018.
- [19] V. S. Dave, B. Zhang, P.-Y. Chen, and M. A. Hasan, "Neural-brane: Neural bayesian personalized ranking for attributed network embedding," *Data Science and Engineering*, vol. 4, no. 2, pp. 119–131, 2019.
- [20] S. Cavallari, V. W. Zheng, H. Cai, K. C.-C. Chang, and E. Cambria, "Learning community embedding with community detection and node embedding on graphs," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2017, pp. 377–386.
- [21] J. J. Choong, X. Liu, and T. Murata, "Learning community structure with variational autoencoder," in *2018 IEEE international conference on data mining (ICDM)*. IEEE, 2018, pp. 69–78.
- [22] S. Souravlas, A. Sifaleras, M. Tsintogianni, and S. Katsavounis, "A classification of community detection methods in social networks: a survey," *International Journal of General Systems*, vol. 50, no. 1, pp. 63–91, 2021.
- [23] G. Agarwal and D. Kempe, "Modularity-maximizing graph communities via mathematical programming," *The European Physical Journal B*, vol. 66, no. 3, pp. 409–418, 2008.
- [24] D. Aloise, S. Cafieri, G. Caporossi, P. Hansen, S. Perron, and L. Liberti, "Column generation algorithms for exact modularity maximization in networks," *Physical Review E*, vol. 82, no. 4, p. 046112, 2010.
- [25] T. N. Dinh and M. T. Thai, "Finding community structure with performance guarantees in complex networks," *arXiv preprint arXiv:1108.4034*, 2011.
- [26] A. Miyauchi and Y. Miyamoto, "Computing an upper bound of modularity," *The European Physical Journal B*, vol. 86, no. 7, p. 302, 2013.
- [27] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014, pp. 701–710.
- [28] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 2016, pp. 855–864.
- [29] X. Huang, J. Li, and X. Hu, "Accelerated attributed network embedding," in *Proceedings of the 2017 SIAM international conference on data mining*. SIAM, 2017, pp. 633–641.
- [30] C. Yang, Z. Liu, D. Zhao, M. Sun, and E. Chang, "Network representation learning with rich text information," in *Twenty-fourth international joint conference on artificial intelligence*, 2015.
- [31] X. Su, S. Xue, F. Liu, J. Wu, J. Yang, C. Zhou, W. Hu, C. Paris, S. Nepal, D. Jin *et al.*, "A comprehensive survey on community detection with deep learning," *IEEE Transactions on Neural Networks and Learning Systems*, 2022.
- [32] D. Zhang, J. Yin, X. Zhu, and C. Zhang, "User profile preserving social network embedding," in *IJCAI International Joint Conference on Artificial Intelligence*, 2017.
- [33] D. Jin, Z. Yu, P. Jiao, S. Pan, D. He, J. Wu, P. Yu, and W. Zhang, "A survey of community detection approaches: From statistical modeling to deep learning," *IEEE Transactions on Knowledge and Data Engineering*, 2021.
- [34] K. G. Dizaji, F. Zheng, N. Sadoughi, Y. Yang, C. Deng, and H. Huang, "Unsupervised deep generative adversarial hashing network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 3664–3673.
- [35] Q. Yuan and B. Liu, "Community detection via an efficient nonconvex optimization approach based on modularity," *Computational Statistics & Data Analysis*, vol. 157, p. 107163, 2021.
- [36] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *International conference on machine learning*. PMLR, 2017, pp. 1263–1272.
- [37] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.
- [38] O. Shchur and S. Günnemann, "Overlapping community detection with graph neural networks," *arXiv preprint arXiv:1909.12201*, 2019.
- [39] D. D. Paepe, D. N. Avendano, and S. V. Hoecke, "Implications of z-normalization in the matrix profile," in *International Conference on Pattern Recognition Applications and Methods*. Springer, 2019, pp. 95–118.
- [40] M. E. Newman, "Analysis of weighted networks," *Physical review E*, vol. 70, no. 5, p. 056131, 2004.
- [41] C. L. Giles, K. D. Bollacker, and S. Lawrence, "Citeseer: An automatic citation indexing system," in *Proceedings of the third ACM conference on Digital libraries*, 1998, pp. 89–98.
- [42] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su, "Arnetminer: extraction and mining of academic social networks," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2008, pp. 990–998.
- [43] A. L. Traud, P. J. Mucha, and M. A. Porter, "Social structure of facebook networks," *Physica A: Statistical Mechanics and its Applications*, vol. 391, no. 16, pp. 4165–4180, 2012.
- [44] L. Danon, A. Diaz-Guilera, J. Duch, and A. Arenas, "Comparing community structure identification," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2005, no. 09, p. P09008, 2005.
- [45] K. Y. Yip, D. W. Cheung, and M. K. Ng, "Harp: A practical projected clustering algorithm," *IEEE Transactions on knowledge and data engineering*, vol. 16, no. 11, pp. 1387–1397, 2004.
- [46] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding," in *Proceedings of the 24th international conference on world wide web*, 2015, pp. 1067–1077.

Location Accuracy of a Ground Station based on RSS in the Rice Channel

Jarosław Michalak

Military University of Technology
 Kaliskiego st. No 2, 00-908 Warsaw, Poland
 E-mail: jaroslaw.michalak@wat.edu.pl

□

Abstract— The article presents the assessment of the potential accuracy of the location of the terrestrial radio signal source in the Rice channel using the received signal filtering and the non-linear regression function. The basic assumption for the parameterization of the channel was to use a drone with a simple antenna system and Received Signal Strength analysis from multiple measurement points (Multilateration location). Preliminary results under Rice-typical channel conditions indicate position estimation errors of the order of 60 meters for $K=7$, which in the assumed network structure is approximately 10% of the actual average distance. By using properly parameterized filtration systems (Kalman algorithm and Moving Average algorithm set), it is possible to increase this accuracy by one-third of the initial value.

I. INTRODUCTION

THE problem of locating the source of radio radiation is very broad. This task can be viewed as a service to a locator or located user. The method of its implementation varies depending, for example, on the type of services (military or civil), type of system, technical capabilities of devices (e.g. antenna set), conditions in which the task is performed (e.g. type of land cover, user traffic characteristics). In the changing world of technology, one of the recently popular branches of development is the area related to the use of unmanned aerial vehicles [1]-[3]. They are of course used for different purposes and hence can be classified differently.

This study assumes the use of simple, small Unmanned Aerial Vehicles (UAV) with an uncomplicated antenna system (omnidirectional antenna with low gain) and limited computing possibilities (e.g. [4]—[5]). These types of UAVs are relatively cheap and can perform various types of services in a team (e.g. a swarm of UAVs). The article considers the operation of a single UAV, which uses Received Signal Strength (RSS) recorded during the flight and the Multilateration method [6] to locate the radio signal source. Location methods based on RSS measurement belong to the Range Based group and are used relatively easily, realizing location with moderate accuracy, which can be increased by increasing the number of bearing points.

Technical details such as the curvature of the Earth resulting from geodetic corrections are omitted in the study.. The results of the analysis shed additional light on the

potential averaged efficiency of using simple, single UAVs in the process of locating the radiation source based on a blind flight over the area of network operation, when the UAV performs other functions simultaneously [7].

Chapter 2 is a reference to the state of knowledge in the analyzed area. The following chapters describe the network structure (Chapter 3), the radio channel model (Chapter 4), the location method and filtering algorithms for the channel response (Chapter 5), and the results of simulation analyzes (Chapter 6).

II. RELATED WORKS

The use of UAVs to locate the source of radio radiation, both indoor and outdoor, is currently the subject of several research and applications. Depending on the goals to be achieved and the technical capabilities of tracking systems and radiation sources, various types of location methods are proposed in [8]—[13]. Others are used in military applications, others in civil applications.

In a situation where advanced technical support from ground nodes cannot be expected, the possibility of using a simple and easy to implement location method based on the measurement of the RSS is often considered (e.g. [14]—[17]). Such algorithms, in connection with the situation when we have a large number of measurement values (the UAV carries out a relatively large number of measurements during the flight while performing other tasks (e.g. securing the network integrity [7], [18]) may offer increased location accuracy, by using the so-called Multilateration algorithms (unlike Trilateration, when the location is based on information from only 3 measurement points [19].

When assessing the possible location accuracy, it should be taken into account that the channel distortions on the Air to Ground connections depend mainly on the type of terrain [20], however, it can be assumed that due to the elevation of the UAV above the ground level, signal fading is not very deep [21] and we often deal with a Rice channel with a coefficient from $K = 7$ to $K = 14$. Nevertheless, increasing the accuracy of the location requires additional filtration operations, the most popular of which are Kalman filtration [22] and the so-called Moving Average.

III. THE NETWORK STRUCTURE

From the point of view of the ground station location algorithm itself, the structure of the network does not matter much. However, to cover the issue in more practical reality,

□ This work was financed by the Military University of Technology under research project no. UGB/22-744/2022/WAT on "Modern radio technologies in military communication systems".

it was assumed that the task of the UAV, which performs the network flight and records signals from terrestrial [7] radio sources, is to indicate the location of nodes beyond the network coverage (not connected, Fig. 1) to undertake subsequent actions to connect them with the rest of the network (e.g. by sending retranslation drones). To average the results, tests were carried out for 4 different route cases, differing in the value of the parameter R , which can be interpreted as the turning radius of the UAV.

The following main assumptions were made about such a network:

1. The terrestrial network is relatively stationary
2. Network nodes do not support UAV measurements (they do not have GPS receivers).

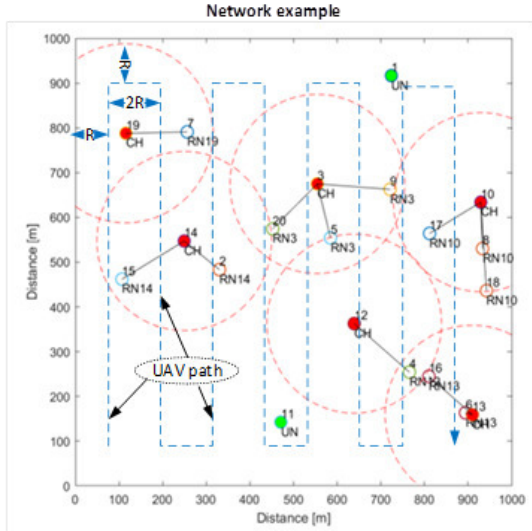


Fig. 1 An example of the location of 20 nodes of a clustered network. RN - Regular Node, CH - Cluster Head (red), UAV - Unmanned Aerial Vehicle (blue line), UN - Unknown Node (green, the disconnected node)

IV. THE RADIO CHANNEL MODEL

The characteristics of the radio channel for the case considered in the material (ground-air link of UAV) mainly depend on the following factors:

1. Drone flight altitudes.
2. UAV flight speed and associated Doppler shift.
3. Land cover (city, suburban area, mountainous area, etc.).
4. Network operating frequencies.
5. Volatility of the above-mentioned factors.

An overview of these types of channels can be found e.g. in [20]. Additional discussion of this type of propagation conditions, e.g. in [18]. The conditions assumed in this study assume an outdoor LOS (Line of Sight) link modelled by the Rice radio channel. According to the guidelines contained in [21], the K factor characterizing the depth of decays ranges from 7 to 14.

We can write down that for the Rice channel:

1. K factor being the ratio of the power of the direct ray to the value of the sum of the powers of the other reflected components

$$K = \frac{\vartheta^2}{2\delta^2}. \quad (1)$$

2. Ω factor as total received power

$$\Omega = \vartheta^2 + 2\delta^2. \quad (2)$$

The amplitude of the received signal is defined as:

$$\vartheta^2 = \frac{K}{1+K} \Omega \text{ and } \delta^2 = \frac{\Omega}{2(1+K)}; \quad (3)$$

and we write the probability density function as:

$$f(x) = \frac{2(K+1)x}{\Omega} \exp\left(-K - \frac{(K+1)x^2}{\Omega}\right) I_0\left(2\sqrt{\frac{K(K+1)}{\Omega}}x\right). \quad (4)$$

Exemplary realizations of the amplitude of the signal received by the UAV for the values of the coefficients $K = 7$ during a single flight over the network (along the selected route) are shown in Fig. 2. The model did not take into account the Doppler shift and antenna characteristics, assuming low UAV speed and omnidirectional characteristics of the antenna with zero gain.

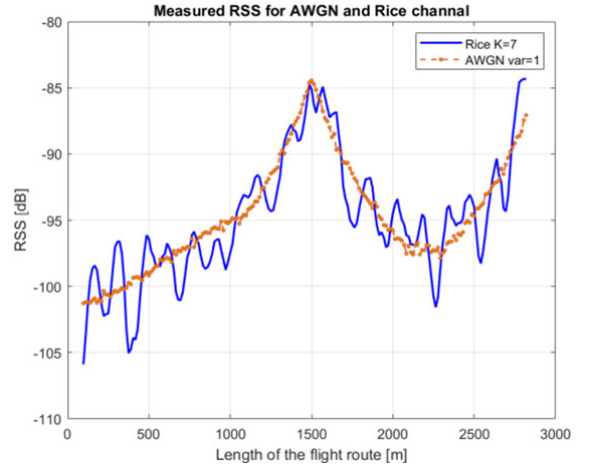


Fig. 2 The amplitude of the signal in the Rice channel for the coefficient $K = 7$

V. DESCRIPTION OF THE LOCATION METHOD

As mentioned in Chapter 2, you can find many literature sources describing different kinds of localization methods. However, if we accept the above-mentioned organizational and technical constraints imposed on network and UAV devices, the problem narrows significantly. In addition, it is worth noting that the assessment of the accuracy of the location in the Rice channel presented in this article is based on the assumption that we have a large set of measurement data, which results in the use of Multilateration algorithms [23], [6], Fig. 3.

Having the bearings from many points, we can save the estimated distance from the localized point in the form:

$$d_i^2 = (x_i - x)^2 + (y_i - y)^2, \quad (i = 1, 2, \dots, n), \quad (5)$$

where:

d – estimated distance from the localized point,
 x_i – coordinate x of DF no. i ,
 y_i – coordinate y of DF no. i .

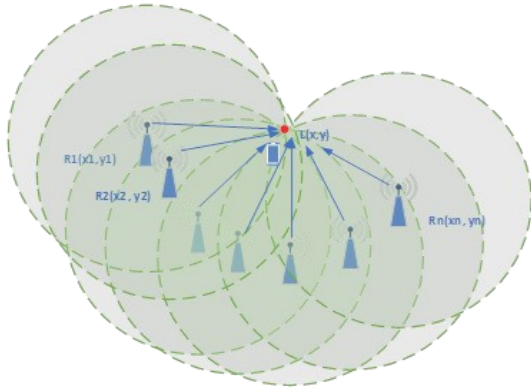


Fig. 3 An example of a directional finding using the Multilateration method with the use of n Directional Finders (DF). $L(x, y)$ - position of the localized station

The position of the signal source can be expressed, for example, in a general form as a solution to the algebraic equation [24]:

$$\begin{bmatrix} 1 & -2x_1 & -2y_1 & -2z_1 \\ 1 & -2x_2 & -2y_2 & -2z_2 \\ 1 & -2x_3 & -2y_3 & -2z_3 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & -2x_n & -2y_n & -2z_n \end{bmatrix} \begin{bmatrix} x^2 + y^2 + z^2 \\ x \\ y \\ z \end{bmatrix} = \begin{bmatrix} d_1^2 - x_1^2 - y_1^2 - z_1^2 \\ d_2^2 - x_2^2 - y_2^2 - z_2^2 \\ d_3^2 - x_3^2 - y_3^2 - z_3^2 \\ \vdots \\ d_n^2 - x_n^2 - y_n^2 - z_n^2 \end{bmatrix} \quad (6)$$

where z - coordinate z (in a spatial case).

Here, to determine the estimate of the radio signal source location, the MATLAB nonlinear regression function was used, giving the model function in the form $\text{modelfun} = @(b, X) (\text{abs}(b(1) - X(:, 1)).^2 + \text{abs}(b(2) - X(:, 2)).^2).^(1/2)$ and the starting point of searching for the optimal solution (position of the target station) in the center of the monitored plane.

To increase the accuracy of the location, the measured signal was pre-filtered. For comparison, Kalman filtering and 7 different Moving Average algorithms were used (Fig. 4).

In addition, before performing the proper location calculation, taking into account the fact that the filtered signal is subject to a time shift (delay), a corresponding correcting time shift and averaging window for the Moving Average methods were made individually for each filter result.

This action was aimed at minimizing the vector distance between the distorted original signal and the filtered signal according to the relationship:

$$\text{MinDistance} = \min \left| \sum_{i=1}^n \sqrt{R_i^2 + Y_i^2} - \sqrt{R_i^2 + Yf_i^2} \right|, \quad (7)$$

where:

- R_i – i -th point of the UAV route,
- Y_i - i -th RSS level of measured signal,
- Yf_i - i -th RSS level of filtered signal.

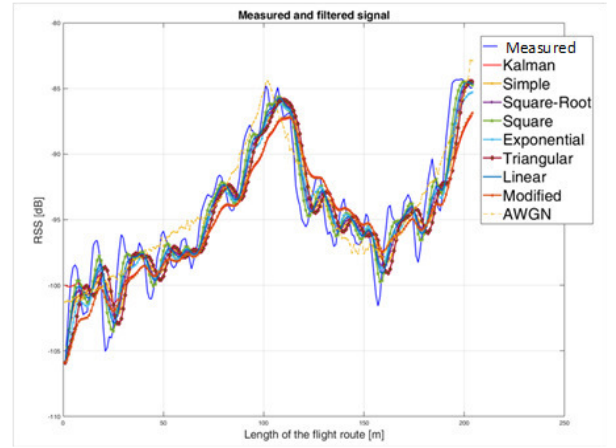


Fig. 4 The effect of applying filtration using various algorithms

This action resulted in a significant reduction in tracking errors.

VI. SIMULATION RESULTS

I. System parametrization

The simulations were carried out in the MATLAB R2021b simulation environment (9.11.0.1769968) using the parameters presented in Table . 30 channel conditions randomization was performed for each set value of the R and K parameters.

TABLE I.
SIMULATION PARAMETERS

Parameter	Value
Channel models	AWGN, Rice
K factor of Rice channel	7 to 14
Noise Variance	1
Number of UAV	1
The number of random repetitions of each case	30
Number of measures	In the range from 144 to 362 depending on the UAV route.
R [m]	100, 150, 200, 250

II. Results

The results of the simulation tests are shown in Fig.5 and Fig.6. It can be initially concluded that each of the filtration methods increases the efficiency of localization to a similar degree. The final error, in the analyzed cases, is the smaller the smaller the value of the K coefficient (channel with deeper fades), and for $K = 7$ it is about one-third smaller than the original value (Fig. 5).

The highest potential in the discussed distortion filtration process, in the range of small K values, is shown by Kalman filtration and the Simple Moving Average.

VII. CONCLUSION

The article assesses the accuracy of the location of the radio ground station in the Rice Channel based on RSS level measurements by the UAV flying around the entire network deployment area. As a result of the application of the Multilateration method and the nonlinear regression function

supported by the filtering algorithms of the measured signal, the location accuracy was improved by several to several dozen meters depending on the depth of the decays (the greater the effect, the deeper the value of the decays). The further direction of work will be related to research using other types of channels and verification of the obtained results in real channel conditions.

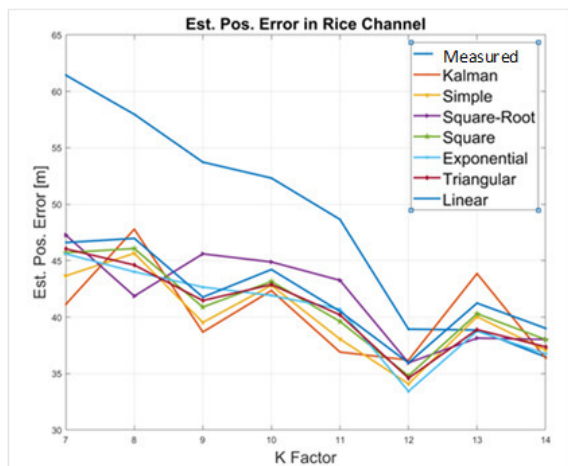


Fig. 5 Mean position error for Rice channels and various measurement signal filtering methods

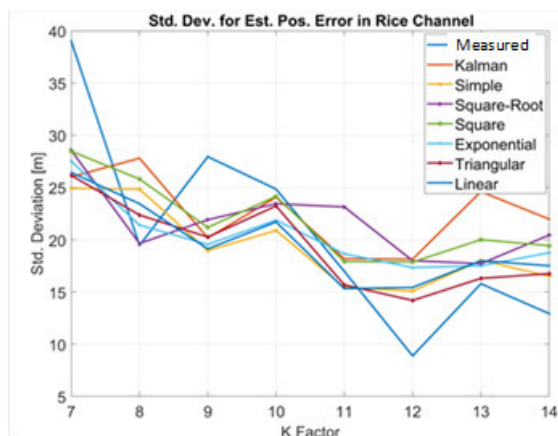


Fig. 6 The standard deviation for the mean position error in the Rice channels and different filtration methods

REFERENCES

- [1] H. Wang, H. Zhao, J. Zhang, D. Ma, J. Li, J. Wei, „Survey on Unmanned Aerial Vehicle Networks: A Cyber Physical System Perspective,” *IEEE Communications Surveys & Tutorials*, pp. 1027-170, vol.22, No.2, 2020
<http://dx.doi.org/10.1109/COMST.2019.2962207>.
- [2] T. D. Chuyen, H. V. Huy, T. L. Nguyen, „Control design of an UAV-Q based on feedback linearization and optimum modulus methods,” w *Proceedings of the Sixth International Conference on Research in Intelligent and Computing*, 2021, <http://dx.doi.org/10.15439/2021R20>.
- [3] T. Adam, F. Babič, „UAV Mission Definition and Implementation for Visual Inspection,” w *Proceedings of the 16th Conference on Computer Science and Intelligence Systems*, 2021, <http://dx.doi.org/10.15439/2021F24>.
- [4] N. Boonyathanming, S. Gongmenee, P. Kayunyeam, P. Wutticho, S. Prongnuch, „Design and Implementation of Mini-UAV for Indoor Surveillance,” *International Electrical Engineering Congress*, 10-12 March 2021, <http://dx.doi.org/10.1109/IEECON51072.2021.9440350>.
- [5] S. Gupte, P.I.T. Mohondas, J.M. Conred, „A Survey of Quadrotor Unmanned Aerial Vehicles,” *IEEE Xplore*, 2012, <http://dx.doi.org/10.1109/SECOn.2012.6196930>.
- [6] C.D. Morales, D. Guevara, M.A. Truvol, „Using Terrestrial Radio Links to Multilateration Techniques,” *IEEE COLCOM*, 2016, <http://dx.doi.org/10.1109/ColComCon.2016.7516389>.
- [7] J. Michalak, „Efficiency of selected drone flight algorithms in increasing the level of ad-hoc network connectivity without knowing the location of disconnected nodes,” *37th International Business Information Management Conference (IBIMA)*, May 2021.
- [8] S. Goswami, *Indoor Location Techniques*, Milpitas: Springer, 2013, <http://dx.doi.org/10.1007/978-1-4614-1377-6>.
- [9] Ed. H.A. Karimi, *Advanced Locatin-Based Technologies and Services*, London: CRC Press, 2013.
- [10] Ed. S.A.Zekavat, R.M. Buehrer, *Handbook of Position Location*, Hoboken: Wiley, 2012.
- [11] Y. Liu, Z. Yang, *Locatin, Localization and Localizabiility*, London: Springer, 2011, <http://dx.doi.org/10.1007/978-1-4419-7371-9>.
- [12] R.A. Poisel, *Electronic Warfare Target Location Methods*, Boston, London: Artech House, 2012.
- [13] F. Zafari, A. Gkelias, K.K. Leung, „A Survey of Indoor Localization Systems and Technologies,” *IEEE Communications Surveys & Tutorials*, 2019, <http://dx.doi.org/10.1109/COMST.2019.2911558>.
- [14] S. Uluscan, T. Filik, „A Survey on the Fundamentals of RSS based Localization,” 2016, <http://dx.doi.org/10.1109/SIU.2016.7496069>.
- [15] N. Saeed , H. Nam, T. Y. Al-Naffouri , M. S. Alouini, „A State-of-the-Art Survey on Multidimensional Scaling-Based Localization Techniques,” *IEEE Communications Surveys & Tutorials*, Vol.21 2019, <http://dx.doi.org/10.1109/COMST.2019.2921972>.
- [16] S. Bohidar, S. Behera, C. R. Tripathy, „A Comparative View on Received Signal Strength (RSS) Based location Estimation in WSN,” *IEEE International Conference on Engineering and Technology (ICETECH)*, March\ 2015, <http://dx.doi.org/10.1109/ICETECH.2015.7275032>.
- [17] Q.Duy, P.De, „A Survey of Fingerprint-Based Outdoor Localization,” *IEEE Communications Surveys & Tutorials*, 2016, <http://dx.doi.org/10.1109/COMST.2015.2448632>.
- [18] J. Michalak, „Location accuracy of a radio ground station in the Rice channel by the multilateration method with the use of non-linear regression function,” *39th International Business Information Management Conference (IBIMA)*, April 2022.
- [19] O.S. Oguejiofor, A.N. Aniedu, H.C., Ejiolor, A.U. Okolibe, „Trilateration Based localization Algorithm for Wireless Sensor Network,” *International Journal of Science and Modern Engineering*, Vol.1, Issue 10 September 2013.
- [20] W. Khawaya, I. Guvenc, D.W. Matolak, U.C. Fiebig, N. Schneckenburger, „A Survey of Air-to-Ground Propagation Channel Modeling for Unmanned Aerial Vehicles,” *IEEE Communicaitons Surveys & Tutorials*, Vol.21, No.3 2019, <http://dx.doi.org/10.1109/COMST.2019.2915069>.
- [21] Y.Diang, Y. Xiao, J. Xie, T. Zhang, „A Time-varying Transition Channel Model for Air-Ground Communication,” *IEEE Xplore*, 2017, <http://dx.doi.org/10.1109/DASC.2017.8102055>.
- [22] M.S. Grewal, A.P. Andrews, „Kalman Filtering, Theory and Practice Using MATLAB,” by *John Wiley & Sons, Inc.*, 2015.
- [23] M.N. Rahman, I.A.T. Hanuranto, R. Mayasari, „Trilateration and Iterative Multilateration Algorithm for Localization Schemes on Wireless Sensor Network,” *International Conference on Control, Electronics, Renewable Energy and Communications*, 2017, <http://dx.doi.org/10.1109/ICCEREC.2017.8226710>.
- [24] A. Norrdine, „An Algebraic Solution to the Multilateration Problem,” *International Conference on Indoor Positioning and Indoor Navigation*, 13-15 November 2012.

1st Workshop on Complex Networks: Theory and Application

IN the nature and the world around us, we can observe many network structures that interconnect various elements such as cells, people, urban centers, network devices, companies, manufacturing machines, etc. Most of them have the nature of evolving networks whose structure changes over time. The analysis of such systems from the complex networks point of view allows for better understanding of the processes within them, which can be used to optimize their structure, improve their management methods, detect failures, improve their operating efficiency and plan their development and evolution.

The main goal of this event is to exchange knowledge and experience between specialists from different areas who in their research and design work use theories and solutions characteristic for complex systems. We believe that the meeting will create new ideas and concepts that will affect the development of contemporary methods of design, operation and analysis of network systems.

TOPICS

The list of topics includes, but is not limited to:

- Complex networks architecture
- Large scale networks analytics
- Mathematical and numerical analysis of networks
- Modeling of computer networks
- Cognitive networks
- Visualizations of network processes
- Dynamics on networks
- Biological and physical models on networks
- Dynamic modification of communication protocols parameters for enterprise and ISP systems
- Complex network management
- Performance modeling and analysis in complex networks
- Network function virtualization

- Social networks
- Graph theory and network algorithm application
- Evolving networks
- Detection of anomalies in the functioning of an enterprise-class computer network element
- Predictive maintenance
- Network technologies supporting society 5.0 and education 5.0
- Architecture for next-generation network applications
- Distributed complex systems for remote working and collaboration
- Algorithms for controlling and monitoring complex computer networks

TECHNICAL SESSION CHAIRS

- **Bolanowski, Marek**, Rzeszow University of Technology, Poland
- **Paszkiewicz, Andrzej**, Rzeszow University of Technology, Poland

PROGRAM COMMITTEE

- **Al-Naday, Mays**, University of Essex, United Kingdom
- **Ballas, Rüdiger G.**, Mobile University of Technology, Germany
- **Houssein, Essam H.**, Minia University, Egypt
- **Ignaciuk, Przemysław**, Lodz University of Technology, Poland
- **Kryvyi, Serhii**, Taras Shevchenko National University of Kyiv, Ukraine
- **Kuchanskyy, Vladislav**, Institute of Electrodynamics of the National Academy of Sciences of Ukraine
- **Palau, Carlos**, Universitat Politècnica de València, Spain
- **Provotar, Oleksandr**, Taras Shevchenko National University of Kyiv, Ukraine

The effectiveness analysis of selected IT tools for predictions of the Covid-19 pandemic

Paweł Dymora, Mirosław Mazurek, Kamil Łyczko
Rzeszów University of Technology
al. Powstańców Warszawy 12, 35-959 Rzeszów, Poland
Email: pawel.dymora@prz.edu.pl, mirekmaz@prz.edu.pl,
160773@stud.prz.edu.pl

Abstract—The article presents the problem of the complexity of prediction and the analysis of the effectiveness of selected IT tools in the example of the Covid-19 pandemic data in Poland. The study used a variety of tools and methods to obtain predictions of extinct infections and mortality for each wave of the Covid-19 pandemic. The results are presented for the 4th wave with a detailed description of selected models and methods implemented in the prognostic package of the statistical programming language R, as well as in the Statistica and Microsoft Excel programs. Naive methods, regression models, exponential smoothing methods (including ETS models), ARIMA models, and the method of artificial intelligence - autoregressive models built by neural networks (NNAR) were used. Detailed analysis was performed and the results for each of these methods were compared.

I. INTRODUCTION

PREDICTION is the process of making certain anticipations with a definable probability of how phenomena will develop in the future. It derives from the field of statistics. It is rational in nature and uses data from the past in its course. Forecasting is used in various areas, such as predicting the weather, electricity consumption, product prices, etc. The resulting forecasts can provide valuable information and help in making decisions about future activities.

Predicts makes it possible to determine the possible future values of a time series. Various methods are available for their determination. They are based on mathematical models that describe the values of the series. The models may take into account many factors, e.g. historical values of the series, values of predictors, characteristics of the series, etc. A model is created by performing a series analysis and parameter estimation based on the data at hand.

Since the beginning of the Covid-19 pandemic, many disease prediction models have emerged, shaping the interest of the media, policymakers, and the broader public [1, 2]. However, forecasting the future development of a pandemic is challenged by the inherent uncertainty rooted in many "unknown unknowns," not only about the contagious virus itself, but also the human, social, and political factors that coevolve and keep the future of the pandemic open. Fore-

casting models have varying degrees of predictive accuracy. Researchers have attempted analyses during previous epidemics of the 21st century, namely SARS, H1N1, and Ebola. As reported in [1, 3], predictions of Ebola deaths have often been far from the ultimate reality, with a strong tendency to overestimate. It is therefore important to communicate the uncertainty of such analyses. The Covid-19 pandemic spread rapidly around the world, and researchers attempted to estimate the risk. Many researchers around the world have used various prediction techniques such as the Susceptible-Infected-Recovered model, Susceptible-Exposed-Infected-Recovered model, and Automatic Regressive Integrated Moving Average (ARIMA) model to predict the spread of this pandemic. The ARIMA technique has not been extensively used in Covid-19 forecasting by researchers due to the claim that it is not suitable for use in complex and dynamic contexts. However, in [4], the authors proposed the use of time series algorithms, Autoregressive Integrated Moving Average (ARIMA) and Autoregressive (AR). ARIMA-based models showed promising results compared to AR-based models. However, the most difficult challenge was parameter identification due to the sudden increase-decrease trend in coronavirus cases. The proposed work presents prediction quality scoring metrics for both models. In [5], verification was performed to see how accurate the best-fit predictions of the ARIMA model were with the actual values reported after the entire prediction period. The results showed that despite the dynamic nature of the disease and the continuous changes made by the Kuwaiti government, the actual values for most of the observed period were within the prediction limits of our chosen ARIMA model with a 95% confidence interval. Another direction taken by the researchers is to apply machine learning (ML) based forecasting mechanisms. ML models have long been used in many application domains that required the identification and prioritization of adverse threat factors. In the paper [6], four standard prediction models such as linear regression (LR), least absolute shrinkage and selection operator (LASSO), support vector

machine (SVM), and exponential smoothing (ES) were used to predict risk factors. The results proved that ES performs best among all the models used, followed by LR and LASSO, which perform well in predicting new confirmed cases, mortality, and recovery rate, while SVM performs poorly in all prediction scenarios given the available data set.

The author's paper [7] proposes an approach to forecasting the spread of pandemics based on a vector autoregression model. Time series of the number of new cases and the number of new deaths were combined to obtain a common prediction model. Test results based on data from the United Arab Emirates, Saudi Arabia, and Kuwait showed that the proposed model achieved a high level of accuracy, outperforming many existing methods, which can be a valuable tool in pandemic management.

The pandemic has shown that having knowledge and prediction of its spread will allow one to respond appropriately and attempt to undertake containment. Results obtained in [8] using a network model based on long-short-term memory (LSTM) have shown promise. The proposed model was used to predict the dates when other countries would be able to contain the spread of Covid-19.

In [9], the authors presented the results of a study on developing a neural network model for predicting the spread of Covid-19. They proposed a predictor based on a classical approach with a deep architecture that learns using the NAdam training model. Official data from government and open source repositories were used for training. The results of the proposed model showed high accuracy, which reached more than 99% in some cases.

The scope of this paper is to use different forecasting methods and to compare the results obtained for a given set of data from the course of the Covid-19 pandemic in Poland. The aim is to analyze the correctness and degree of fit of the forecasts obtained using different algorithms.

In the analysis, we used data on the daily incidence and death rates registered in Poland during the Covid-19 pandemic. On their basis, mathematical models were created, and then forecasts of subsequent values for time horizons of different lengths were carried out. In this way, possible scenarios for the course of the pandemic were presented.

The paper is divided into five chapters. The introduction provides a review of the literature and recent trends used in the prediction of Covid-19 pandemic trends. Chapter 2 characterizes the methods and IT tools used in the study such as regression and ARIMA models, methods based on neural networks, and main packages in R, Statistica, and Excel environments. Chapter 3 presents the selected issue of forecasting the evolution of a pandemic. Chapter 4 presents a description of the experiments and a comparative analysis of the results of prediction methods for the 4th wave of the pandemic in Poland. The summary, conclusion, and scope of future research are presented in Chapter 5.

II. PREDICTION METHODS AND IT TOOLS

The paper presents various techniques used in time series prediction - from simple ones, such as naive methods, through adaptive models and autoregressions to more advanced ones, such as ARIMA models or neuro-networks. The obtained models will be evaluated in terms of their fit to historical realizations of the series and the quality of generated forecasts. In addition, the use of selected software tools useful in forecastings such as Microsoft Excel, Statistica, and RStudio environment using R language developed for statistical purposes is presented. We use regression models which is a statistical method of describing utilizing a function the dependence of the values of some variables (explanatory) on the values of others (explanatory, predictors) [10], and autoregressive (AR) models describe the explanatory variable as a function of its lagged values [11-12]. Prediction is also possible based on neural network-based models. Artificial neural networks have a layered structure, which includes neurons that in a simplified way mimic the operation of cells found in the human brain. With the use of neural networks, it is possible to model the autoregression of time series. It consists in feeding the network input with delayed values of time series. An example of such solution implementation is `nnetar()` function from FORECAST package where one-way neural networks with one hidden layer (NNAR models) are used therefore forecasting [13-15].

III. FORECASTING THE EVOLUTION OF A PANDEMIC USING VARIOUS IT TOOLS AND METHODS

The subject of this study is data related to the course of the Covid-19 pandemic in Poland. The decision to use the results from their timeliness and availability at the time of work creation. The collected data describe the daily numbers of infections, deaths, and tests performed between 05.03.2020 and 25.10.2021. They form a time series that will be used to test the effectiveness of different prediction methods.



Fig. 1 Regression charts of a series of infection numbers for the 4th wave

TABLE I.
A SUMMARY OF THE VALUES OF THE MEASURES OF THE ACCURACY OF THE PREDICTIONS OF THE NUMBER OF INFECTIONS OBTAINED BY THE NAIVE METHODS

		ME	MAE	MSE	RMSE	MAPE
Forecasts for wave 1	simple naive method	61.03333	109.1	16914.433	130.0555	14.77851
	seasonal naive method	75.56667	108.2333	16411.967	128.1092	14.6654
	incremental naive method	-3.80327	111.4168	16287.586	127.6228	16.56339
Forecasts for wave 2	simple naive method	10850.97	10850.97	144349654	12014.56	49.82617
	seasonal naive method	12019.67	12019.67	170837763	13070.49	56.40277
	incremental naive method	10222.1	10222.1	128708885	11344.99	46.87582
Forecasts for wave 3	simple naive method	12524.6	12767.6	215657209	14685.27	48.94015
	seasonal naive method	7015.067	7351.933	72519578	8515.843	29.32963
	incremental naive method	12074.23	12400.53	205532085	14336.39	47.49426
Forecasts for wave 4	simple naive method	1525.667	1593.467	4984209.6	2232.534	53.85878
	seasonal naive method	1719.233	1719.233	5393828.5	2322.462	58.65325
	incremental naive method	1500.687	1571.495	4871069	2207.05	53.14973

To simplify the code of scripts, functions were prepared to create graphs using methods provided by the ggplot2 package of R language. To validate the quality of models and forecasts, functions were created to calculate the values of error measures (ME, MAE, MSE, RMSE, MAPE) and to select the objects of models and forecasts that are characterized by the most satisfactory features (Table I) [12-15].

The fitting of models to historical data and forecasts generated by them are presented in Fig. 1. Figure shows the fit of the finally selected models to the training series and a comparison of the generated forecasts for the 30-day horizon with the test series. The forecast charts also show the ranges for the 80% and 95% confidence levels.

IV. TIME SERIES ANALYSIS OF PANDEMIC WAVE 4 DEATH NUMBERS

The following methods were selected to forecast the values of death numbers for the 4th wave (time period from 26.09.2021 to 25.10.2021):

- R - simple naive method,
- R - seasonal naive method,
- R - incremental naive method,
- R - linear regression model including linear trend and seasonal variables built on training time series with observations from 17.07.2021 to 25.09.2021),

- R - ETS(A,Ad,A) model estimated using historical values of death numbers from 05.03.2020 to 25.09.2021,
- R - ARIMA($2,1,2$)($0,1,1$) model built on the basis of modified by adding constant 1 and undergoing Box-Cox transformation training time series with observations from 05.03.2020 to 25.09.2021,
- R - NNAR($22,1,12$) model selected based on the training series with observations from 05.03.2020 to 25.09.2021,
- Statistica - Holt model estimated based on the training series with observations from 28.03.2020 to 25.09.2021,
- Statistica - Winters model estimated based on modified (addition of constant 1 and natural logarithmization) training time series with values from 28.03.2020 to 25.09.2021.
- Statistica - ARIMA($3,1,3$)($2,1,2$) model built based on modified by adding constant 1 and undergoing natural logarithmization training time series with observations from 28.03.2020 to 25.09.2021,
- Microsoft Excel - ETS(A,A,A) model estimated from the modified training series with death numbers from 28.03.2020 to 25.09.2021.

We concluded that predictions obtained using the Winters method in the Statistica program are characterized by very large values in comparison with predictions created by other methods. They differed significantly from the test (actual) values and were therefore not considered in further analyses. The predictions (excluding those mentioned) are shown in the graph (Fig. 2).

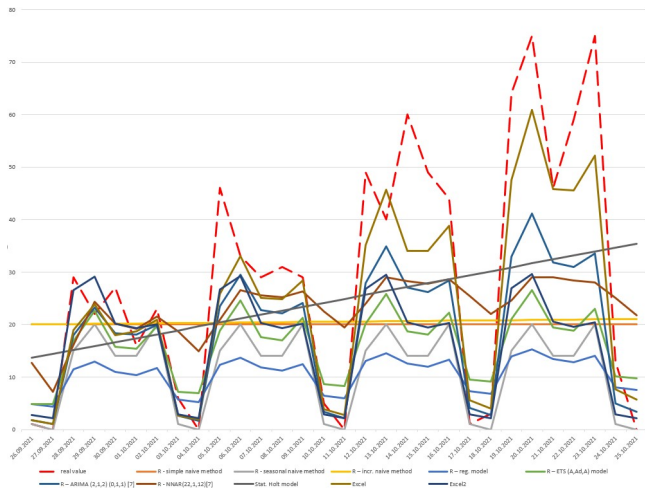


Fig. 2 Comparison of predictions with actual death counts for the 4th wave

Ex post forecast error measures were calculated. The calculation of MAPE values omitted cases for which the value of observations in the test series was equal to 0 (27.09.2021, 04.10.2021, 11.10.2021, and 25.10.2021). The most satisfactory predictions were obtained from the ARIMA(3,1,3) (2,1,2) model in Statistica software.

V. CONCLUSION

The study used a variety of tools and methods to obtain projections of expired infection and death rates for each wave of the Covid-19 pandemic. Forecasting was performed using models and methods implemented in the forecast package of the statistical programming language named R and the programs Statistica and Microsoft Excel. With the first tool, naive methods, regression models, exponential smoothing methods (including ETS models), ARIMA models, and artificial intelligence method - autoregressive models built by neural networks (NNAR) were applied for the examined time series. In Statistica software, modules were used to create predictions based on exponential smoothing methods and ARIMA models. In the case of Excel, predictions were obtained using prediction sheets.

MAE, RMSE, and MAPE error measures were used to verify the quality of the applied models and methods. Based on their values calculated for the training time series, a preliminary selection of the best variants of methods and models within one category was made (e.g. selection of the best regression model from among models taking into account

various predictors). The final selection of the best solution for a particular time series was based on the values of the ex-post forecast error measures (calculated for the test set). Based on the analysis of the final results (forecasts obtained with the selected methods), based on the values of the test error measures, the solutions that allowed obtaining the most satisfactory predictions were indicated. For the time series of infection rates, a multiplicative variant of the Winters method from the family of exponential equalization methods was selected as the best forecasting method. For the time series of deaths, the use of ARIMA models was selected as the best approach.

REFERENCES

- [1] Le Ha Anh and Nguyen Minh Trang and Nguyen Thi Phuong Linh, The Influence of Work-from-home on job performance during COVID-19 pandemic: Empirical evidence Hanoi, Vietnam, Proceedings of the International Conference on Research in Management & Technovation, vol. 28, pp. 73-81, <http://dx.doi.org/10.15439/2021KM59>, 2021
- [2] F. Grabowski, A. Paszkiewicz, M. Bolanowski: Wireless networks environment and complex networks, Lecture Notes in Electrical Engineering, Analysis and Simulation of Electrical and Computer Systems, Springer International Publishing Switzerland, ISBN 978-3-319-38545-7, vol. 324, str. 261-270, 2015.
- [3] P. Nadella, A. Swaminathan, S. V. Subramanian, SV, "Forecasting efforts from prior epidemics and COVID-19 predictions". EUROPEAN JOURNAL OF EPIDEMIOLOGY, Vol. 35, Issue 8, Page 727-729, DOI: 10.1007/s10654-020-00661-0, 2020.
- [4] D. Prajapati, M. Kanojia, "Forecasting of COVID-19 Cases in INDIA Using ARIMA and AR Time-Series Algorithm", Proceedings of the 13th International Conference on Soft Computing and Pattern Recognition (SOCPAR 2021), Book Series Lecture Notes in Networks and Systems, Vol. 417, Page 361-370, DOI: 10.1007/978-3-030-96302-6_33, 2022.
- [5] G. Alabdulrazzaq, M. N. Alenezi, Y. Rawajfih, B. A. Alghannam, A. A. Al-Hassan, F. S. Al-Anzi, "On the accuracy of ARIMA based prediction of COVID-19 spread", RESULTS IN PHYSICS, Vol. 27, Article Number 104509, DOI: 10.1016/j.rinp.2021.104509, 2021.
- [6] F. Rustam, A. A. Reshi, A. Mehmood, S. Ullah, B. W. On, W. Aslam, G. S. Choi, "COVID-19 Future Forecasting Using Supervised Machine Learning Models", IEEE ACCESS, Vol. 8, Page 101489-101499, DOI: 10.1109/ACCESS.2020.2997311, 2020.
- [7] K. Rajab, F. Kamalov, A. K. Cherukuri, "Forecasting COVID-19: Vector Autoregression-Based Model", Arabian Journal for Science and Engineering, DOI: 10.1007/s13369-021-06526-2, 2022.
- [8] S. Kumar, R. Sharma, T. Tsunoda, T. Kumarevel, A. Sharma, "Forecasting the spread of COVID-19 using LSTM network", BMC BIOINFORMATICS, Vol. 22, Issue SUPPL 6, Article Number 316, DOI: 10.1186/s12859-021-04224-2, 2021.
- [9] M. Wiczorek, J. Silka, M. Wozniak, "Neural network powered COVID-19 spread forecasting model", CHAOS SOLITONS & FRACTALS, Vol. 140, Article Number 110203, DOI: 10.1016/j.chaos.2020.110203, 2020.
- [10] M. Sobczyk, "Prognozowanie. Teoria, przykłady, zadania." Wydawnictwo Placet, 2008.
- [11] R. J. Hyndman, G. Athanasopoulos, "Forecasting: Principles and Practice", <https://otexts.com/fpp2/>. Access 15.09.2021 r.
- [12] <https://www.machinelearningplus.com/time-series/arima-model-time-series-forecasting-python/> Access 27.09.2021 r.
- [13] <https://www.rstudio.com/>. Access 27.09.2021 r.
- [14] <https://www.rdocumentation.org/packages/forecast/versions/8.15>. Access 27.09.2021 r.
- [15] <https://www.statsoft.pl/Programy/Architektura-STATISTICA/Programy-desktop/>. Access 27.09.2021 r.

Denotational Model and Implementation of Scalable Virtual Machine in CPDev

Jan Sadolewski

Department of Computer and Control Engineering
 Rzeszow University of Technology,
 al. Powstańców Warszawy 12, 35-959 Rzeszów, Poland
 Email: js@kia.prz.edu.pl

Bartosz Trybus

Department of Computer and Control Engineering
 Rzeszow University of Technology,
 al. Powstańców Warszawy 12, 35-959 Rzeszów, Poland
 Email: btrybus@kia.prz.edu.pl

Abstract—Denotational semantic model and its implementation in C/C++ are presented for a virtual machine executing programs written in the CPDev development environment according to IEC 61131 standard. Programs written in IEC ST language are compiled to control-oriented intermediate language designed specifically for the machine. Architecture of the machine and its operation are represented by formal semantic model which assigns abstract algebraic objects to denote machine behaviour. Execution of intermediate language instructions is described in details by denotational semantic equations followed strictly by C/C++ implementations to assure reliability of the machine.

I. INTRODUCTION

THE concept of virtual machines as platforms for software execution had a significant impact on computer science for almost half a century [1], [2]. A virtual machine (VM) is understood as a kind of processor with a certain instruction set and data types, which is implemented by software on particular hardware platforms. A VM processes an intermediate code generated by a compiler from a source program. The concept of VMs has been gaining importance due to the widespread use of the Java [3] and the .NET [4], [5]. Solutions based on VMs have some important advantages, namely a) source program and intermediate code are independent of target platforms, b) one compiler is sufficient, c) programs are executed in safe environments. The disadvantages include slower execution of the intermediate code and the need to develop a runtime environment suitable for the target platform.

This paper deals with the development of a runtime environment for control programs written according to the standard IEC 61131 [6]. The IEC standard defines the programming languages: Structured Text (ST), Instruction List (IL), Ladder Diagram (LD), Function Block Diagram (FBD) and Sequential Function Chart (SFC). Here, employing the VM concept appears to be particularly justified in order to cope with the large variety of target platforms. The CPDev engineering environment [7] uses this concept to program controllers according to the IEC standard. It consists of a compiler translating ST to intermediate code and a VM-based runtime system written in C. Initially, small and medium-scale controllers were considered [8]. Recently, however, motivated by applications with extensive calculations, arose the need to extend the CPDev compiler and its VM. Therefore, some additional assumptions were imposed, namely:

- to develop a semantic model of the machine and its intermediate language followed by a C implementation,
- to achieve scalability of the machine depending on the particular hardware and application requirements,

The model formalizes the VM description as an interpreter of the intermediate code, including instruction and operand decoding, and low-level operations while executing the instructions. Denotational semantics [9], [10] appropriate to formally describe programming languages are applied [11], [12]. For denotations the λ -notation is adequate and, therefore, applied [9], [13].

II. VIRTUAL MACHINE ARCHITECTURE

The architecture of the VM includes [14]: code and data memories, stacks and registers. The instruction processing module fetches successive instructions from *Code memory* and executes them acquiring values of operands either from *Data* or *Code memory*. Results are stored in *Data memory*.

Registers: The program counter is kept in the *CodeReg* register. The data base register *DataReg* is set by calls to and returns from subprograms, including function blocks and functions. When entering a subprogram the current values of *CodeReg* and *DataReg* are pushed onto *Code stack* and *Data stack*. The machine also includes the *Flags* register with status flags signaling errors or unusual situations.

VMASM intermediate language: The virtual machine operates as an interpreter of assembly code called VMASM (VM Assembler). The syntax is:

```
[ :label ] instruction [ operand1 ] [, operand2 ] ...
```

Instruction set: Functions and system procedures are two kinds of virtual machine instructions. Examples are shown in Table I.

The compilation of the simple ST instruction

```
MOTOR := (START OR MOTOR) AND NOT STOP;
```

is presented in Listing 1. At first, the variables *START* and *MOTOR* are ORed and the result is stored in a temporary variable ?LR?A03. If it is zero, JZ jumps to :?A02, where MCD sets *MOTOR* to 0 interpreted as FALSE. If ?LR?A03 is not zero, the function NOT performs logical negation of STOP storing the result in ?LR?A05. If it is zero, JZ jumps to the

TABLE I
SELECTED FUNCTIONS AND PROCEDURES

Mnemonic	Meaning	Operator
Functions		
ADD	Addition	+ (arithm.)
SUB	Subtraction	- (arithm.)
GT	Greater	>
EQ	Equal	=
NE	Not equal	<>
NOT	Negation	- (unary)
AND	Logical and	&
System procedures		
JMP	Unconditional jump	
JZ, JNZ	Conditional jumps	
CALB	Subroutine call	
RETURN	Return from subroutine	
MCD	Initialize data	

:?A02 as before to set MOTOR to 0. If not, the first MCD sets MOTOR to 1 (TRUE). This is followed by JMP to :?E08 from which another code begins.

Listing 1. Example of VMASM mnemonic code

```
OR ?LR?A03, START, MOTOR
JZ ?LR?A03, :?A02
NOT ?LR?A05, STOP
JZ ?LR?A05, :?A02
MCD MOTOR, #01, #01
JMP :?E08
:?A02
MCD MOTOR, #01, #00
:?E08
```

III. SCALABILITY

The data types and instructions of the VMASM language are defined in XML-formatted library configuration files (LCF).

Types and instructions: A portion of type definitions is shown in Listing 2. By applying `deny-type` one can restrict some data types. Aliases to existing types and special types not specified in the IEC standard can be defined, too.

Listing 2. Type definition

```
<deny-type name="LREAL" />
<type name="USINT" implement="alias">
  <alias name="BYTE"/>
</type> ...
```

Functions: The definition of one in the group of ADD functions is presented in Listing 3. The virtual machine code `vmcode` consists of two bytes, with the first one 01 identifying the group, whereas the second *2 indicates a flexible number of inputs (*) and identifies the data type (2) processed. The two components of `vmcode` are called group and type identifier, and are denoted by `ig` and `it`. By choosing an appropriate `it`, type-specific functions such as `ADD:SINT`, `ADD:INT`, etc. are defined.

Listing 3. Function definition

```
<function name="ADD" vmcode="01*2" return="INT">
```

```
<operands>
  <op no="*" name="a*" type="INT"/>
</operands>
</function> ...
```

Procedures: All system procedures are identified by `ig=1C`. The second byte `it` of `vmcode` indicates a particular procedure. The definitions of `JNZ` and `CALB` are shown in Listing 4. `JNZ` executes a conditional jump to `:gclabel`, `CALB` calls the subprogram at `:gclabel`.

Listing 4. Procedure definitions

```
<sysproc name="JNZ" vmcode="1C01">
  <op no="0" name="cnd" type="BOOL"/>
  <op no="1" name="clbl" type=":gclabel"/>
</sysproc>
<sysproc name="CALB" vmcode="1C16">
  <op no="0" name="inst" type=":rdlabel"/>
  <op no="1" name="clbl" type=":gclabel"/>
</sysproc>
```

Operand types: The following types are available:

- `:gclabel` `:gdlabel` – global pointer (address) to *Code/Data memory*,
- `:rclabel` `:rdlabel` – address relative to actual content of code/data register,
- `:imm` – immediate value (direct, constant).

IV. SEMANTIC MODEL

Semantic models provide formal descriptions of programming languages [11], [15]. In case of VMASM, the model consists of domains describing the virtual machine's states, memory functions, value interpreters relating memory to VMASM types, limited range operators, and a universal semantic function.

Semantic domains: The domain *BasicTypes* consists of four sets reflecting the memory sizes of the VMASM types. The domain *Address* specifies 16- or 32-bit implementation. The general domain *Memory* is a function mapping *Address* to *Byte1*. *Stack* models a sequence (*) of *Address* domains (Kleene closure).

$$\text{BasicTypes} = \text{Byte1} + \text{Bytes2} + \\ + \text{Bytes4} + \text{Bytes8}$$

$$\text{Address} = \text{if } \text{AddressSize} = 2 \text{ then} \\ \text{Bytes2} \text{ else } \text{Bytes4}$$

$$\text{Memory} = \text{Address} \rightarrow \text{Byte1}$$

$$\text{CodeMemory} = \text{Memory}$$

$$\text{Stack} = \text{Address}^*$$

$$\text{CodeStack} = \text{Stack}$$

$$\text{CodeReg} = \text{Address}$$

$$\text{Flags} = \text{Bytes2}$$

State: The purpose of program execution is to change the current state into a new one. The state of the VM is a Cartesian product of memory domains, stacks, registers and flags, i.e.

$$\text{State} = \text{CodeMemory} \times \text{DataMemory} \times$$

$$\begin{aligned} & \times \text{CodeStack} \times \text{DataStack} \times \\ & \times \text{CodeReg} \times \text{DataReg} \times \text{Flags} \end{aligned}$$

Model functions: The functions presented below model low-level operations executed on memory, stacks and flags.

- Get data from memory (read)

$$\begin{aligned} G1BM &= (\text{Address} \times \text{Memory}) \rightarrow \text{Byte1} \\ G2BM &= (\text{Address} \times \text{Memory}) \rightarrow \text{Bytes2} \\ G4BM &= (\text{Address} \times \text{Memory}) \rightarrow \text{Bytes4} \\ G8BM &= (\text{Address} \times \text{Memory}) \rightarrow \text{Bytes8} \end{aligned}$$

- Get address from memory

$$\text{GetAddress} = (\text{Address} \times \text{Memory}) \rightarrow \text{Address}$$

- Memory update (write)

$$\begin{aligned} U1BM &= (\text{Address} \times \text{Memory} \times \text{Byte1}) \rightarrow \text{Memory} \\ U2BM &= (\text{Address} \times \text{Memory} \times \text{Bytes2}) \rightarrow \text{Memory} \\ U4BM &= (\text{Address} \times \text{Memory} \times \text{Bytes4}) \rightarrow \text{Memory} \\ U8BM &= (\text{Address} \times \text{Memory} \times \text{Bytes8}) \rightarrow \text{Memory} \end{aligned}$$

- Memory move (copy)

$$\text{MemMove} = (\text{Address} \times \text{Memory} \times \text{Address} \times \text{Memory} \times \text{Byte1}) \rightarrow \text{Memory}$$

The two *Addresses* represent source and target, respectively with *Byte1* denoting number of bytes being moved.

- Stack functions

$$\begin{aligned} \text{Push} &= (\text{Stack} \times \text{Address}) \rightarrow \text{Stack} \\ \text{Pop} &= \text{Stack} \rightarrow (\text{Address} \times \text{Stack}) \end{aligned}$$

Value interpreters: The following sample functions provide numerical interpretations of memory chunks.

$$\begin{aligned} \text{BoolOf} &= \text{Byte1} \rightarrow \text{BOOL} \\ \text{FromBool} &= \text{BOOL} \rightarrow \text{Byte1} \\ \text{IntOf} &= \text{Bytes2} \rightarrow \text{INT} \\ \text{FromInt} &= \text{INT} \rightarrow \text{Bytes2} \\ \text{DIntOf} &= \text{Bytes4} \rightarrow \text{DINT} \\ \text{FromDInt} &= \text{DINT} \rightarrow \text{Bytes4} \\ \text{LIntOf} &= \text{Bytes8} \rightarrow \text{LINT} \\ \text{FromLInt} &= \text{LINT} \rightarrow \text{Bytes8} \end{aligned}$$

Limited range operators: The virtual machine executes arithmetic operations in limited ranges, dependent on the particular types. For signed integers addition \oplus is defined by

$$\begin{aligned} a \oplus b &= \text{if } (a + b) \geq 0 \\ &\text{then } (a + b) \bmod (-\text{MinRange}(a)) \\ &\text{else } (a + b) \bmod (-\text{MinRange}(a) + 1) \end{aligned}$$

where *MinRange* for SINT, INT, DINT and LINT means -128 , -32768 , -2^{31} or -2^{63} , respectively. For unsigned integers USINT, UINT etc. we have

$$a \oplus b = (a + b) \bmod (\text{MaxRange}(a))$$

where *MaxRange* means 256, 65536 etc.

Unification: The operator $:=$ used in expression unifies both sides. If the right side is an expression, then the left side is a variable with the value of the right side (assignment). If the left side is a tuple and the right side a variable, then the variable is split into the tuple's components.

Universal semantic function: To jointly express the concept of decoding group and type, followed by execution of a particular instruction, one may define a universal function covering all instructions

$$\mathcal{U}[\text{any_instruction}] = \text{State} \rightarrow \text{State} \quad (1)$$

Internally, after decoding *ig* and *it*, this function calls a specific function of the form

$$\mathcal{C}[\text{instruction}] = \text{State} \rightarrow \text{State} \quad (2)$$

Instruction decoding: Instruction decoding can formally be expressed by the denotational semantic equation shown in Listing 5. According to [9] or [13], the λ -expression has the form of $\lambda s. \text{body}$, where *s* denotes the current state and *body* determines the value returned by the function. The *body* consists of a sequence of operations, the first of which splits current state *s* into a tuple composed of model components. The other operations decode the values of identifiers *ig* and *it*, update the code register to *cr*₂ and, by means of **match ... with** statements, call particular \mathcal{C} functions. The result provided by \mathcal{C} defines the new state *s*₁ returned by the function \mathcal{U} .

Listing 5. Denotational equation for the function \mathcal{U}

$$\begin{aligned} \mathcal{U}[\text{any_instruction}] &= \\ &\lambda s. (\text{cm}, \text{dm}, \text{cs}, \text{ds}, \text{cr}, \text{dr}, \text{flg}) := s \\ &\text{ig} := G1BM(\text{cr}, \text{cm}) \\ &\text{cr}_1 := \text{cr} \oplus 1 \\ &\text{it} := G1BM(\text{cr}_1, \text{cm}) \\ &\text{cr}_2 := \text{cr}_1 \oplus 1 \\ &\text{s}_1 := \text{match ig with} \\ &\quad | 01 \rightarrow \text{match it with} \\ &\quad \quad | 22 \rightarrow \mathcal{C}[\text{ADD:INT:r:op1:op2}] \\ &\quad \quad \quad (\text{cm}, \text{dm}, \text{cs}, \text{ds}, \text{cr}_2, \text{dr}, \text{flg}) \\ &\quad \quad | 32 \rightarrow \mathcal{C}[\text{ADD:INT:r:op1:op2:op3}] \\ &\quad \quad \quad (\text{cm}, \text{dm}, \text{cs}, \text{ds}, \text{cr}_2, \text{dr}, \text{flg}) \\ &\quad \quad | \dots \\ &\quad \quad \text{end} \\ &\quad | \dots \\ &\quad \text{end} \\ &\text{s}_1 \end{aligned}$$

V. DENOTATIONS AND IMPLEMENTATIONS

Denotational equations modeling the VMASM instructions have the common form $\mathcal{C}[\dots] = \lambda s. \text{body}$ where the dots on the left side are replaced by the descriptor of a particular instruction. Splitting current state *s* into components through unification

$$(\text{cm}, \text{dm}, \text{cs}, \text{ds}, \text{cr}, \text{dr}, \text{flg}) := s \quad (3)$$

is the first operation in *body*.

Assume that while calling a particular function \mathcal{C} by the universal function \mathcal{U} , the code register *cr* points to the first operand. (actually *cr*₂ in Listing 5). If the operand is a variable

or label, then its value, i.e. address, is acquired from code memory cm by

$$operand := GetAddress(cr, cm) \quad (4a)$$

In case of a global variable or label, $operand$ stands for a direct address in data or code memory. If, however, the operand is a local variable of a subprogram, then the value $operand$ means an address relative to the current value of data base register dr , which was set earlier by a subprogram call. Therefore, the address of a local variable is obtained by adding

$$operandaddr := dr \oplus operand \quad (4b)$$

The value of a variable, here shown for a Boolean, in data memory dm is read out and interpreted by composition

$$BoolOf(G1BM(operandaddr, dm)) \quad (5)$$

If an instruction has another operand, the code register cr is incremented to point to the next memory location by

$$cr_1 := cr \oplus AddressSize \quad (6)$$

Defining the new state s_1 as the tuple

$$s_1 := (cm, \dots) \quad (7)$$

is the last operation in $body$, with the dots being replaced by new values of the data memory (if updated), stacks, etc.

Basic procedures: The denotational equation of the unconditional jump JNZ is presented in Listing 6 and subprogram call CALB in Listing 7.

Listing 6. Denotation of JNZ procedure

```

C[[JNZ:cnd:clbl]] = λs.
  (cm, dm, cs, ds, cr, dr, flg) := s
  cnd := GetAddress(cr, cm)
  cndaddr := dr ⊕ cnd
  cr1 := cr ⊕ AddressSize
  clbl := GetAddress(cr1, cm)
  cr2 := cr1 ⊕ AddressSize
  ctl := BoolOf(G1BM(cndaddr, dm))
  s1 := match ctl with
    | true → (cm, dm, cs, ds, clbl, dr, flg)
    | false → (cm, dm, cs, ds, cr2, dr, flg) end
s1

```

Listing 7. Denotation of CALB procedure

```

C[[CALB:inst:clbl]] = λs.
  (cm, dm, cs, ds, cr, dr, flg) := s
  inst := GetAddress(cr, cm)
  iad := dr ⊕ inst
  cr1 := cr ⊕ AddressSize
  clbl := GetAddress(cr1, cm)
  cr2 := cr1 ⊕ AddressSize
  s1 := (cm, dm, Push(cs, cr2), Push(ds, dr), clbl, iad, flg)
s1

```

The equation of JNZ has two operands, the conditional variable cnd in data memory and the code label $clbl$ as before. The address $cndaddr$ is determined according to (4a) and (4b) (with the content dr of the data base register equal

to zero in case of a global variable). The code register is incremented to cr_1 to obtain the address $clbl$ and, then, to cr_2 pointing to the next instruction. The Boolean value ctl controlling execution is determined as in (5). Depending on ctl , the code register of s_1 includes either $clbl$ or cr_2 .

The first operand of CALB is the label of an instance in data memory for which the subprogram beginning at the label $clbl$ is executed. The instance address iad and the subprogram address $clbl$ are determined as before. cr_2 points to the next instruction. Since the contents of cr_2 , dr must be remembered for the subprogram return, they are pushed onto corresponding stacks.

Listing 8 shows C implementation of the JNZ and CALB. All system procedures having the common group identifier IC are handled by a single general function IG_SYSPROC_1C, with type identifier it as its parameter. The command switch selects a particular procedure. Each of the code segments sets $codeReg$ to a new value depending on the respective meaning. CALB also modify $dataReg$.

Listing 8. Implementations of JNZ and CALB procedures

```

void IG_SYSPROC_1C(BYTE it) {
  switch(it) {
    case 0x01: /* JNZ conditional jump */ {
      ADDRESS cndaddr = dataReg + GetCodeAddress();
      ADDRESS clbl = GetCodeAddress();
      BOOL ctl = BOOLOf(G1BMData(cndaddr));
      if (ctl) codeReg = clbl;
    } break;
    case 0x16: /* CALB call a function block */ {
      ADDRESS iad = dataReg + GetCodeAddress();
      ADDRESS clbl = GetCodeAddress();
      push_CodeStack(codeReg);
      push_DataStack(dataReg);
      dataReg = iad;
      codeReg = clbl;
    } break;
    ... /* other procedures */
    default: /* unknown code */
      flags |= FAULT;
      break; }
}

```

Selected functions: The semantics of function NOT presented in Listing 9 negates the value stored at $op1$. The addresses $raddr$, $op1addr$ are determined as before, followed by the Boolean value bv obtained as in (5). By means of the function $U1BM$ (Sec. IV) the value at $raddr$ in data memory dm is then updated. That value is determined from bv by the $FromBool$ and **match ... with** construct. um denotes the new state of the data memory. The C implementation of the NOT function presented in Listing 10 corresponds directly to its semantics.

In the case of the function EQ (Listing 11) two LINT operands $op1$, $op2$ are checked for equality. The Boolean value cmp follows from comparison (=) of the LINT numbers determined by $LIntOf$. The updated data memory in s_1 is the result of invoking $U1BM$. The byte stored at $raddr$ is given by $FromBool(cmp)$. The function IG_EQ_12 from Listing 12 implements the comparison EQ for all relevant data types (group) via a parameterized macrodefinition EQ_TYPE.

The value of an operand of a particular TYPE is determined in EQ_TYPE by the function TYPE##Of with given sizeof(TYPE).

Listing 9. Denotation of NOT function

```

C[[NOT:r:op1]] = λs.
  (cm, dm, cs, ds, cr, dr, flg) := s
  r := GetAddress(cr, cm)
  raddr := dr ⊕ r
  cr1 := cr ⊕ AddressSize
  op1 := GetAddress(cr1, cm)
  opladdr := dr ⊕ op1
  cr2 := cr1 ⊕ AddressSize
  bv := BoolOf(G1BM(op1addr, dm))
  um := U1BM(raddr, dm, FromBool(match bv with
  | true → false
  | false → true end))
  s1 := (cm, um, cs, ds, cr2, dr, flg)
s1

```

Listing 10. Implementation of NOT function

```

ADDRESS raddr = dataReg + GetCodeAddress();
ADDRESS opladdr = dataReg + GetCodeAddress();
BOOL bv = BOOLOf(G1BMData(opladdr));
U1BM(raddr, FromBOOL(
  bv ? FALSE : TRUE ));

```

Listing 11. Denotation of EQ function

```

C[[EQ:LINT:r:op1:op2]] = λs.
  (cm, dm, cs, ds, cr, dr, flg) := s
  r := GetAddress(cr, cm)
  raddr := dr ⊕ r
  cr1 := cr ⊕ AddressSize
  op1 := GetAddress(cr1, cm)
  opladdr := dr ⊕ op1
  cr2 := cr1 ⊕ AddressSize
  op2 := GetAddress(cr2, cm)
  op2addr := dr ⊕ op2
  cr3 := cr2 ⊕ AddressSize
  cmp := LIntOf(G8BM(op1addr, dm))
  = LIntOf(G8BM(op2addr, dm))
  s1 := (cm, U1BM(raddr, dm,
  FromBool(cmp)), cs, ds, cr3, dr, flg)
s1

```

Listing 12. Implementation of EQ function

```

#define EQ_TYPE(TYPE) \
case IT_EQ_##TYPE & 0x000F: {\
  ADDRESS raddr = dataReg + GetCodeAddress(); \
  ADDRESS opladdr = dataReg + GetCodeAddress(); \
  ADDRESS op2addr = dataReg + GetCodeAddress(); \
  TYPE op1 = TYPE##Of(GetMemData(opladdr, sizeof(TYPE)\
  )); \
  TYPE op2 = TYPE##Of(GetMemData(op2addr, sizeof(TYPE)\
  )); \
  BOOL cmp = op1 == op2; \
  U1BM(raddr, FromBOOL(cmp)); \
break;

void IG_EQ_12(BYTE it) {
  switch (it & 0x0F) {
    EQ_TYPE(SINT);
    EQ_TYPE(INT);
    EQ_TYPE(DINT);

```

```

EQ_TYPE(LINT);
... /* other types */
default: /* unknown code */
  flag |= FAULT;
}
return; }

```

VI. CONCLUSION

Architecture, VMASM intermediate language, and denotational semantic model have been presented for a virtual machine executing IEC control programs. The machine's scalability covers the address size, available data types and instructions. A semantic model was developed to enhance software quality. Denotational equations modeling VMASM instructions are directly followed by C implementations.

ACKNOWLEDGMENT

This project is financed by the Minister of Education and Science of the Republic of Poland within the "Regional Initiative of Excellence" program for years 2019–2022. Project number 027/RID/2018/19, amount granted 11 999 900 PLN.

REFERENCES

- [1] B. Venners, *Inside the Java Virtual Machine*. McGraw-Hill Companies, 1997.
- [2] R. F. Stärk, J. Schmid, and E. Börgen, *Java and the Java Virtual Machine*. Berlin: Springer Heidelberg, 2001.
- [3] T. Lindholm, F. Yellin, G. Bracha, and A. Buckley, *The Java® Virtual Machine Specification*. Oracle America, Inc., 2013.
- [4] T. L. Thai and L. H., *.NET Framework Essentials*. O'Reilly Media, 2001.
- [5] ECMA-335 Standard, *Common Language Infrastructure (CLI)*. Geneva: ECMA, 2012.
- [6] IEC 61131-3 Standard, *Programmable Controllers. Part 3. Programming languages*. International Standard: IEC, 2013.
- [7] D. Rzońca, J. Sadolewski, A. Stec, Z. Świder, B. Trybus, and L. Trybus, "Developing a multiplatform control environment," *JAMRIS*, vol. 13, no. 4, p. 73–84, 2019. [Online]. Available: <https://doi.org/10.14313/JAMRIS/4-2019/40>
- [8] B. Trybus, "Development and Implementation of IEC 61131-3 Virtual Machine," *Theoretical and Applied Informatics*, vol. 23, no. 1, pp. 21–35, 2011.
- [9] M. Gordon, *The Denotational Description of Programming Languages*. New York: Springer-Verlag, 1979.
- [10] J. Stoy, *Denotational Semantics: The Scott–Strachey approach to programming language theory*. Massachusetts: Massachusetts Institute of Technology, 1979.
- [11] K. Slonneger and B. L. Kurtz, *Formal Syntax and Semantics of Programming Languages: A Laboratory-Based Approach*. Addison-Wesley Publishing Company, Inc, 1995.
- [12] N. S. Papaspyrou, "Denotational semantics of ANSI C," *Computer Standards & Interfaces*, vol. 23, pp. 169–185, 2001. doi: 10.1016/S0920-5489(01)00059-9
- [13] H. Barendregt and E. Barendsen, *Introduction to Lambda Calculus*. online: <ftp://ftp.cs.ru.nl/pub/CompMath.Found/lambda.pdf>, 2000.
- [14] J. Sadolewski and B. Trybus, "Compiler and virtual machine of a multiplatform control environment," *Bulletin of the Polish Academy of Sciences: Technical Sciences*, vol. 70, no. 2, p. e140554, 2022. doi: 10.24425/bpasts.2022.140554
- [15] D. Schmidt, *Denotational Semantics: A Methodology for Language Development*. Kansas State University, Manhattan: Department of Computing and Information Sciences, 1997.

Room mapping system using RFID and mobile robots

Mariusz Skoczylas

Department of Electronic and Telecommunication Systems
Rzeszow University of Technology,
al. Powstańców Warszawy 12, 35-959 Rzeszów, Poland
Email: msko@prz.edu.pl

Łukasz Gotówko

Department of Electrodynamics and Electrical Machine Systems
Rzeszow University of Technology,
al. Powstańców Warszawy 12, 35-959 Rzeszów, Poland
Email: l.gotowko@prz.edu.pl

Mateusz Salach

Department of Complex Systems
Rzeszow University of Technology,
al. Powstańców Warszawy 12, 35-959 Rzeszów, Poland
Email: m.salach@prz.edu.pl

Bartosz Trybus

Department of Computer and Control Engineering
Rzeszow University of Technology,
al. Powstańców Warszawy 12, 35-959 Rzeszów, Poland
Email: btrybus@prz.edu.pl

Marcin Hubacz

Department of Computer and Control Engineering
Rzeszow University of Technology,
al. Powstańców Warszawy 12, 35-959 Rzeszów, Poland
Email: m.hubacz@prz.edu.pl

Bartosz Pawłowicz

Department of Electronic and Telecommunication Systems
Rzeszow University of Technology,
al. Powstańców Warszawy 12, 35-959 Rzeszów, Poland
Email: barpaw@prz.edu.pl

Abstract—The article presents a prototype indoor space mapping solution using RFID transponders. The autonomous robot reads the information they contain using a set of several readers, which improves the process. The design of the robot prototype is based on the STM32 NUCLEO module. Two types of transponder grids are considered, square and triangular. Simulation results for both grid types show the efficiency of reading information from transponders by the moving robot.

Unfortunately, reading the information does not provide knowledge about the location of the object on a surface or in space. Therefore, quickly reaching the correct object can be a problem. Therefore, in the field of RFID systems, the problem of mapping the space in which transponders are located is an important research area [12], [13], [14], [15], [16], [17], [18].

I. INTRODUCTION

AUTONOMOUS navigation robots are currently the subject of much research, and the RFID technique [1], [2], [3] is often used in experimental developments as part of a robot navigation and indoor mapping system [4], [5]. Such systems are used to identify objects with passive or semi-passive tags equipped with a writable memory with an information capacity far superior to the currently used barcodes [1]. RFID systems can be used in many areas, particularly in the ISM field (Industrial, Scientific, Medical). Building exploration, surface, and space mapping using such robots is also a promising area of research [6], [7]. Complex systems consisting of a robot group can also pose communication challenges [8], [9].

A great advantage of RFID systems is that the information can be read from multiple tagged objects simultaneously. A robot that is equipped with an RFID reader can use information and coordinates stored in RFID transponders embedded in walls, floors, doors, and furniture [10], [11]. This is illustrated in Fig. 1.

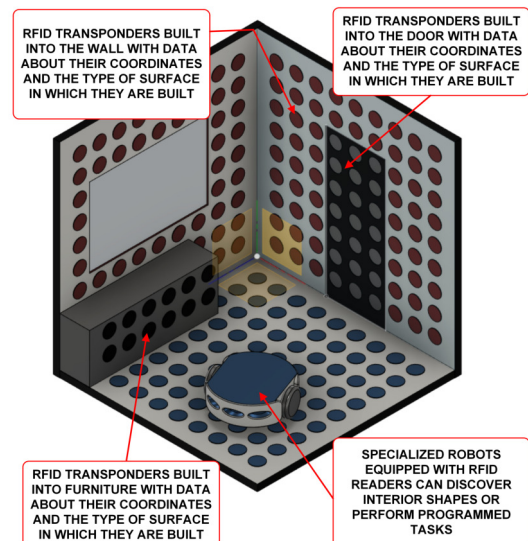


Fig. 1: The use of RFID transponder grids for indoor mapping with a robot

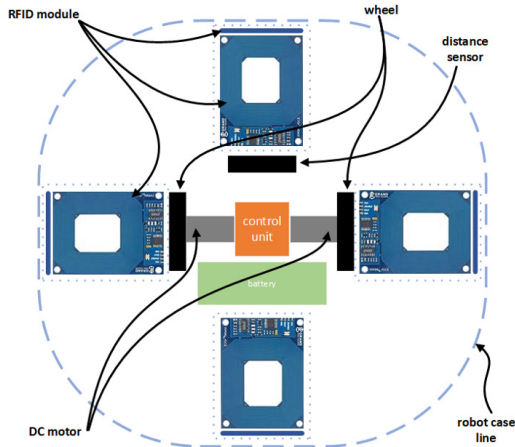


Fig. 2: An overview of the autonomous robot

It has also been noted that it is possible to use this technique in a different way and attempt to determine the position of a mobile robot equipped with an RFID reader based on the readings and analysis of the signals obtained when reading information from tags placed at known locations [19], [20], [21], [22]. Many of the existing studies are based on the use of the RSSI (received signal strength index) method [22], [23]. This is sometimes extended with methods related to odometry and the use of EKF filtering to increase the accuracy of position estimation [24]. There are also original solutions based, for example, on the phase measurement of the signal reflected from the transponder and carrying the information read from the transponder [25].

However, the above-mentioned studies mainly cover a very obtuse mathematical apparatus and almost completely ignore the issue of location support with data stored in the transponder memory. This gap was very quickly recognized, which resulted in the development of various solutions using RFID transponders to determine the location and orientation of a robot equipped with an RFID reader, starting from simple solutions with information on how the robot should read the stored transponders [26] to advanced systems using measurement of the power of the signal reflected from RFID transponders placed in a grid [27].

II. PROPOSED SOLUTION

This work considers a room mapping system using a robot with four readers and the components shown schematically in Fig. 2. The nature of RFID systems must consider the fact that uninterrupted operation of readers is necessary to provide power to transponders deployed in space and to correctly read information from their memory.

Due to the principle of operation, it is possible to distinguish two types of solutions that determine the energy transmission in RFID systems. The first are systems operating on the principle of inductive coupling, and the second is propagation coupling. The energy transferred between the reader and the transponder is transmitted through a magnetic field (Fig. 3),

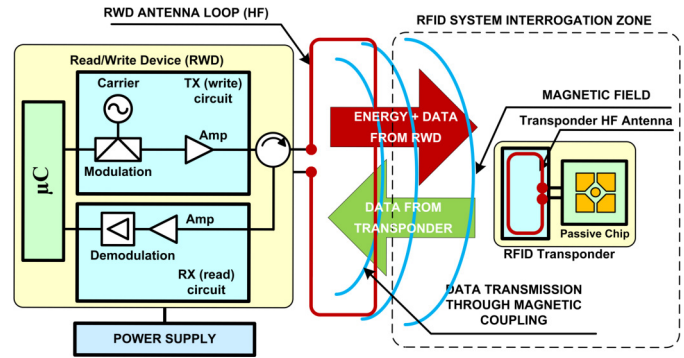


Fig. 3: General scheme of the RFID identification system

and its amount depends on the surface and mutual position of the receiving and transmitting antennas. For proper operation of the system, it is necessary to activate the RFID transponder antenna with resonant frequency, because it causes maximal current flow in the antenna circuit.

Passive transponders are mostly used with inductive coupling. In the basic scenario each transponder transmits its serial number as long as it is in the area of correct reader operation. The maximum range is achieved when the magnetic field lines generated by the reader antenna are perpendicular to the winding plane of the badge antenna coil. If the field lines are parallel to the coil, the connection does not occur, and no power is provided. The maximum range of readers is mostly limited by administrative restrictions on the maximum allowable intensity of the magnetic field produced by the reader antenna.

Propagation coupling is used for communication in the UHF band. Here, in contrast to inductive coupled systems, the far-field region is used, and it is assumed that the electromagnetic field strength is independent of the presence of transponders in the field of the reader antenna. The information exchange between transponders and a reader is based on the modulation of magnetic field modulation.

The performance of an RFID system can be comprehensively described by the idea of interrogation zone. It describes field, energy and communication issues of RFID system components [28]. Proper estimation of this area allows a wider implementation of multiple object identification. Due to the different modes of operation, the interrogation zone is defined differently for inductive and propagation systems. However, in both cases, energy is transferred through radio waves, so each must comply with acceptable radiation standards based on CEPT/ERC 70-03 recommendations. Based on these guidelines, the minimum field strength needed to power the transponders can be determined [29]. Like any electrical/electronic equipment, a robot is a potential source of electromagnetic interference and must be designed in accordance with the requirements of EMC Directive 2014/30/EU. There are no subject standards for this class of equipment yet, so the evaluation must be done according to the requirements of the general standard IEC 61000-6-3 [29].

III. RFID TRANSPONDER ARRANGEMENT

Thanks to the use of a grid of RFID transponders (sensors), it is possible to build an environment that will determine the coordinates of the sensors, memorize parameters of the environment and the motion path of moving objects. Thus, this approach makes it possible to implement a control system for an autonomous mobile object [30]. The distribution of transponders, the correlation of distances between them and the area of correct operation of the RFID system significantly affects the number of simultaneously detected transponders. A larger number of transponders in the interrogation zone requires a longer time needed for transponder recognition (use of multiple identification algorithm, reading of data contained in memory of individual transponders). On the other hand, the time influences, to a large extent, parameters of movement of a mobile object, and mainly determines its movement speed. However, temporary lack of transponders in the interrogation zone causes, that the mobile navigation system loses for some time the actuality of position data.

Among the many possibilities for the deployment of transponders in the considered area, two rational solutions can be distinguished as shown in Fig. 4 with triangle and square distribution of transponders. If the activity areas of individual transponders have a circular shape (with radius R), according to the theoretical premises of total area coverage [31], the most economical solution is to locate transponders in the vertices of an equilateral triangle (with side length: $R\sqrt{3}$). However, the problem is more complicated, because this arrangement is superimposed by the area of correct operation of the RFID system (Interrogation Zone - IZ), which is usually approximated to the shape of a circle of radius R [32].

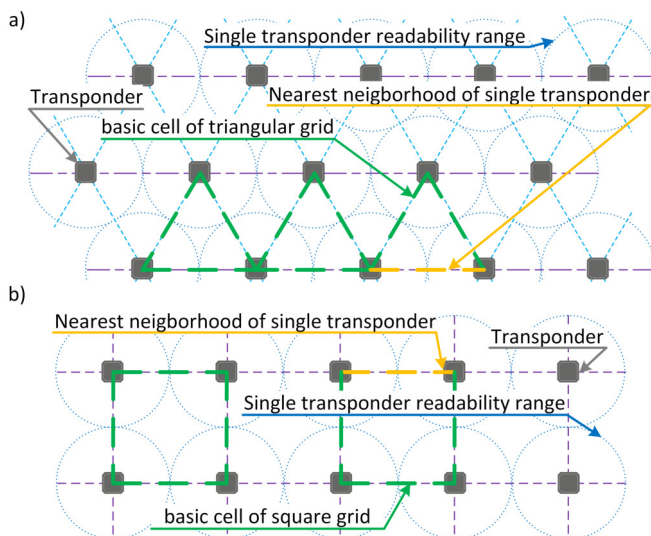


Fig. 4: The structure of transponders deployment: a) triangle, b) square

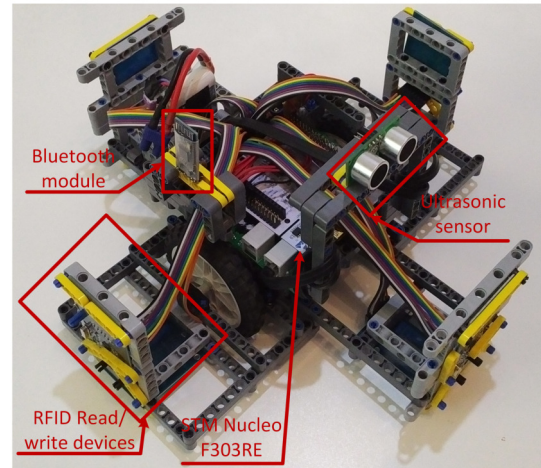


Fig. 5: Perspective view of the robot prototype

IV. THE MOBILE ROBOT

The robot built in this work is intended to be a mobile structure moving inside a building, i.e., in a closed and highly confined space. In order to reduce the likelihood of potential problems during the construction and use of the RFID system, the support structure was made entirely of materials that do not cause wave interference during RFID reader operation, i.e., plastic. A central STM32 NUCLEO-F303RE microprocessor unit with additional electronic components, such as a position sensor and a Bluetooth module, was placed on the support structure in the central part of the robot. The RFID readers were mounted in the robot, two per side in two different planes, horizontally and vertically, respectively. Additionally, a distance sensor has been placed at the front of the robot to detect and avoid obstacles. A Lipo battery pack is used as the power source for the robot.

For primary structure a LEGO mechanical components and drive assemblies were used to build the robot. The drive system consists of two servos. Each of them has an internal reduction gear, so their maximum torque is 8 N/cm and their maximum holding torque is 12 N/cm. In addition, the accuracy of the internal rotation sensor is 1 degree. On the prepared drive base, a skeleton of the supporting structure was mounted, to which the remaining components were then attached, such as: ultrasonic distance sensor, Bluetooth module eight RFID readers (Fig. 5); and indirectly through a special extension overlay: multiplexer, two motor control systems, microprocessor system, 9DoF inertial navigation unit.

V. RESULTS

An application in MathCad was developed for the computer simulation of selected structures of the distribution of transponders on a surface. Two types of structures were analyzed: a square grid and a triangular grid. The sizes of the surface on which the transponders are distributed can be freely defined, but since the structures in question are regular,

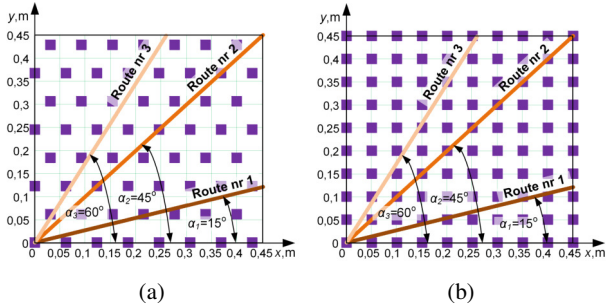


Fig. 6: Assumptions to the simulation process for grid configurations: a) triangle, b) square

the analysis of a limited area of this surface is sufficient for simulation purposes. This is shown in Figure 6.

For comparison purposes, identical areas of $0.45\text{m} \times 0.45\text{m}$ were assumed for both grid configurations. The distances between transponders, respectively for the triangular grid D_T and the square grid D_K , can be defined independently for both cases, but they are in close relation to each other. The case where for both structures, for any antenna position of the reader-programmer system with respect to the transponders, there is at least one transponder in the interrogation zone was chosen as the most reasonable one. The cases omitted are, in which, during movement, the control system does not detect any transponder and must use more advanced algorithms to correctly determine the location. The above assumption leads to the relationship:

$$D_T = 3/2 * \sqrt{3/2} * D_K \quad (1)$$

Considering relation (1), $D_K = 0.05\text{m}$ was assumed, while $D_T = 0.092\text{m}$. The mobile object can move in any direction and according to any defined trajectory, but for simplicity of calculations, it was assumed that the movement will take place only along a straight line, for three different directions (Fig. 6), respectively: Route No. 1, Route No. 2 and Route No. 3). The shape of the region of correct operation was assumed to be a circle with a defined radius R (Fig. 7). The x-axis was discretized by running 100 equidistant lines, and the steps of analyzing individual motion trajectories correspond to their projections on the x-axis, for successive discretization steps. It follows from this that the scope of the analysis of the successive steps is limited to the shortest projection of the motion trajectory on the x-axis. This is the projection of Route 3 and determines the range of analysis in the interval $\langle 0; 0.26 \text{ m} \rangle$.

In order to illustrate the problem in question, below is an analysis of a selected case of configuration of spatial distribution of tags in correlation with the size of the interrogation zone of the RFID system. Fig. 8 compares the number of transponders in the interrogation zone for the selected structures of their distribution: square grid and triangular grid, respectively, when changing the position of the RWD system antenna according to the assumed motion trajectories.

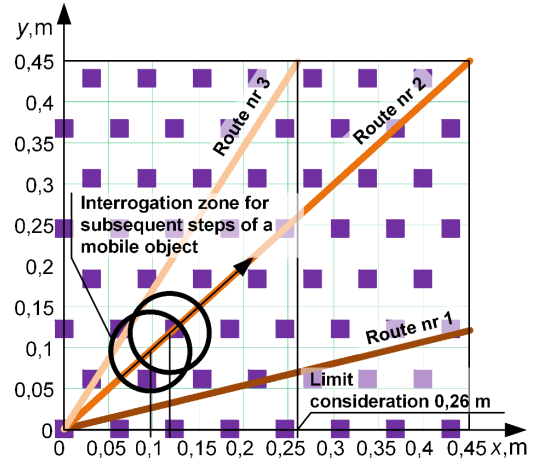


Fig. 7: Interrogation zone and limit consideration

The structure of transponder distribution has a significant influence on the number of transponders, located in the interrogation zone, as well as the dynamics of changes of this number. For comparison purposes, a statistical analysis was performed according to the relation:

$$S_{r_{KwTr}} = \frac{\sum l_{IdKw_k} - l_{IdTr_k}}{l_{IdKw_k}} * 100 \quad (2)$$

where: k is the step of the mobile object (interrogation zone) along the x-axis, l_{IdKw} - number of transponders in the interrogation zone of the RFID system for a square grid, l_{IdTr} - number of transponders in the interrogation zone of the RFID system for a triangular grid. The number of transponders is greater for the square grid, on average by more than 30%. For example: for road 1: $S_{r_{KwTr}} \approx 33\%$, for road 2: $S_{r_{KwTr}} \approx 35\%$, and for road 3: $S_{r_{KwTr}} \approx 29\%$. The

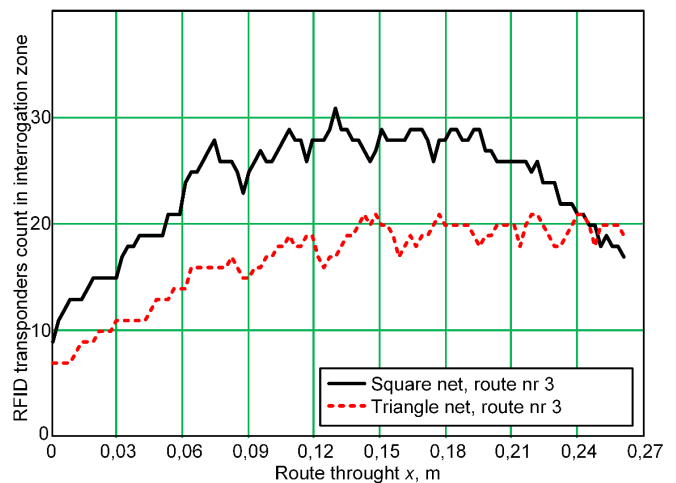


Fig. 8: RFID transponders count in interrogation zone with radius $R = 0.15\text{m}$

observed, rapid increase in the number of transponders in the first phase of traffic (slope of characteristics for road x up to about 0.12m) is caused by reaching the point where there is a full coverage of the IZ into the zone marked with transponders.

The above conclusions lead to the conclusion that the higher resolution of the transponders with respect to the interrogation zone, results in a percentage of more information for a square grid than for a triangular grid. This situation significantly affects the need to process more data in order to decide the next step in the movement of the mobile object, and thus requires an increase in the computing power of the control system or a reduction in the maximum speed of movement of the mobile object.

ACKNOWLEDGMENT

This project is financed by the Minister of Education and Science of the Republic of Poland within the “Regional Initiative of Excellence” program for years 2019–2022. Project number 027/RID/2018/19, amount granted 11 999 900 PLN.

REFERENCES

- [1] K. Finkenzerler, *RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification*, 2nd ed. Wiley Publishing, 2003. ISBN 0470844027
- [2] S. ISO/IEC 14443-3, *Identification cards - Contactless integrated circuit cards - Proximity cards*. International Standard: ISO/IEC, 2016.
- [3] S. ISO/IEC 15693, *Identification cards - Contactless integrated circuit cards - Vicinity cards*. International Standard: ISO/IEC, 2010.
- [4] T. D. Chuyen, R. V. Hoa, N. D. Dien, and T. L. Nguyen, “Mobile robots interacting with obstacles control based on artificial intelligence,” in *Proceedings of the 2021 Sixth International Conference on Research in Intelligent and Computing*, V. K. Solanki and N. H. Quang, Eds., vol. 27, 2021, p. 13–16.
- [5] M. Baïou, A. Quilliot, L. Aduane, A. Mombelli, and Z. Zhu, “Algorithms for the safe management of autonomous vehicles,” in *Proceedings of the 16th Conference on Computer Science and Intelligence Systems*, M. Ganzha, L. Maciaszek, M. Paprzycki, and D. Ślęzak, Eds., vol. 25, 2021, p. 153–162.
- [6] M.-S. Jian and J.-S. Wu, “Rfid applications and challenges,” in *Radio Frequency Identification*, M. B. I. Reaz, Ed. Rijeka: IntechOpen, 2013, ch. 1.
- [7] T. Sanpetchuda and L.-o. Kovavisaruch, “A review of rfid localization: Applications and techniques,” vol. 2, pp. 769 – 772, 06 2008. doi: 10.1109/ECTICON.2008.4600544
- [8] F. Grabowski, A. Paszkiewicz, and M. Bolanowski, *Wireless networks environment and complex networks*. Switzerland: Springer International Publishing, 2015, vol. 324. ISBN 978-3-319-38545-7
- [9] M. Hubacz, B. Pawłowicz, and B. Trybus, “Exploring a surface using rfid grid and group of mobile robots,” in *Automation 2018*, R. Szewczyk, C. Zieliński, and M. Kaliczynska, Eds. Cham: Springer International Publishing, 2018. ISBN 978-3-319-77179-3 pp. 490–499.
- [10] C. Li, L. Mo, and D. Zhang, “Review on uhf rfid localization methods,” *IEEE Journal of Radio Frequency Identification*, vol. 3, no. 4, pp. 205–215, 2019. doi: 10.1109/JRFID.2019.2924346
- [11] S. Willis and S. Helal, “Rfid information grid for blind navigation and wayfinding,” in *9th IEEE International Symposium on Wearable Computers, ISWC 2005*. IEEE, 2005. doi: 10.1109/ISWC.2005.46. ISBN 0769524192 pp. 34–37.
- [12] G. Yang and J. Saniie, “Sight-to-sound human-machine interface for guiding and navigating visually impaired people,” *IEEE Access*, vol. 8, pp. 185 416–185 428, 2020. doi: 10.1109/ACCESS.2020.3029426
- [13] H.-Y. Yu, J.-J. Chen, and T.-R. Hsiang, “Design and implementation of a real-time object location system based on passive rfid tags,” *IEEE Sensors Journal*, vol. 15, pp. 1–1, 09 2015. doi: 10.1109/JSEN.2015.2432452
- [14] C.-Y. Yao and W.-C. Hsia, “An indoor positioning system based on the dual-channel passive rfid technology,” *IEEE Sensors Journal*, vol. 18, no. 11, pp. 4654–4663, 2018. doi: 10.1109/JSEN.2018.2828044
- [15] Q. Wen, Y. Liang, C. Wu, A. Tavares, and X. Han, “Indoor localization algorithm based on artificial neural network and radio-frequency identification reference tags,” *Advances in Mechanical Engineering*, vol. 10, no. 12, p. 1687814018808682, 2018. doi: 10.1177/1687814018808682
- [16] J. Pomarico-Franquiz and Y. Shmaliy, “Accurate self-localization in rfid tag information grids using fir filtering,” *IEEE Transactions on Industrial Informatics*, vol. 10, 02 2014. doi: 10.1109/TII.2014.2310952
- [17] S. P. Subramanian, J. Sommer, S. Schmitt, and W. Rosenstiel, “Ril — reliable rfid based indoor localization for pedestrians,” in *2008 16th International Conference on Software, Telecommunications and Computer Networks*, 2008. doi: 10.1109/SOFTCOM.2008.4669483 pp. 218–222.
- [18] C. Jiang, Y. He, X. Zheng, and Y. Liu, “OmniTrack: Orientation-aware rfid tracking with centimeter-level accuracy,” *IEEE Transactions on Mobile Computing*, vol. 20, no. 2, pp. 634–646, 2021. doi: 10.1109/TMC.2019.2949412
- [19] L. Yang, J. Cao, W. Zhu, and S. Tang, “Accurate and efficient object tracking based on passive rfid,” *IEEE Transactions on Mobile Computing*, vol. 14, no. 11, pp. 2188–2200, 2015. doi: 10.1109/TMC.2014.2381232
- [20] S. S. Saab and Z. S. Nakad, “A standalone rfid indoor positioning system using passive tags,” *IEEE Transactions on Industrial Electronics*, vol. 58, no. 5, pp. 1961–1970, 2011. doi: 10.1109/TIE.2010.2055774
- [21] A. Motroni, P. Nepa, A. Buffi, and B. Tellini, “Robot localization via passive uhf-rfid technology: State-of-the-art and challenges,” in *2020 IEEE International Conference on RFID (RFID)*, 2020. doi: 10.1109/RFID49298.2020.9244884 pp. 1–8.
- [22] A. Motroni, A. Buffi, and P. Nepa, “A survey on indoor vehicle localization through rfid technology,” *IEEE Access*, vol. 9, pp. 17 921–17 942, 2021. doi: 10.1109/ACCESS.2021.3052316
- [23] C. Wu, B. Tao, H. Wu, Z. Gong, and Z. Yin, “A uhf rfid-based dynamic object following method for a mobile robot using phase difference information,” *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–11, 2021. doi: 10.1109/TIM.2021.3073712
- [24] C. Röhrig, D. Heß, and F. Künemund, “Constrained kalman filtering for indoor localization of transport vehicles using floor-installed hf rfid transponders,” in *2015 IEEE International Conference on RFID (RFID)*, 2015. doi: 10.1109/RFID.2015.7113081 pp. 113–120.
- [25] B. Tao, H. Wu, Z. Gong, Z. Yin, and H. Ding, “An rfid-based mobile robot localization method combining phase difference and readability,” *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 3, pp. 1406–1416, 2021. doi: 10.1109/TASE.2020.3006724
- [26] C. Wu, B. Tao, H. Wu, Z. Gong, and Z. Yin, “A uhf rfid-based dynamic object following method for a mobile robot using phase difference information,” *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–11, 2021. doi: 10.1109/TIM.2021.3073712
- [27] C. Jiang, Y. He, X. Zheng, and Y. Liu, “Orientation-aware rfid tracking with centimeter-level accuracy,” in *2018 17th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, 2018. doi: 10.1109/IPSN.2018.00057 pp. 290–301.
- [28] P. Jankowski-Mihuowicz, W. Kalita, and B. Pawłowicz, “Problem of dynamic change of tags location in anticollision rfid systems,” *Microelectronics Reliability*, vol. 48, no. 6, pp. 911–918, 2008. doi: https://doi.org/10.1016/j.microrel.2008.03.006 Thermal, Mechanical and Multi-physics Simulation and Experiments in Micro-electronics and Micro-systems (EuroSimE 2007).
- [29] S. IEC 61000-6-3, *Electromagnetic Compatibility - Part 6-3: Generic Standards - Emission Standard For Residential, Commercial And Light-industrial Environments*. International Standard: IEC, 2020.
- [30] M. Hubacz, B. Pawłowicz, and B. Trybus, “Using multiple rfid readers in mobile robots for surface exploration,” in *Automation 2019*, ser. Advances in Intelligent Systems and Computing, R. Szewczyk, C. Zieliński, and M. Kaliczynska, Eds., vol. 920. Springer, 2019. doi: 10.1007/978-3-030-13273-6_56 pp. 608–615.
- [31] R. Kershner, “The number of circles covering a set,” *American Journal of Mathematics*, vol. 61, no. 3, pp. 665–671, 1939. [Online]. Available: <http://www.jstor.org/stable/2371320>
- [32] M. Konieczny, B. Pawłowicz, J. Potencki, and M. Skoczyła, “Application of rfid technology in navigation of mobile robot,” in *2017 21st European Microelectronics and Packaging Conference (EMPC) Exhibition*, 2017. doi: 10.23919/EMPC.2017.8346907 pp. 1–4.

6th Workshop on Internet of Things—Enablers, Challenges and Applications

THE Internet of Things is a technology which is rapidly emerging the world. IoT applications include: smart city initiatives, wearable devices aimed to real-time health monitoring, smart homes and buildings, smart vehicles, environment monitoring, intelligent border protection, logistics support. The Internet of Things is a paradigm that assumes a pervasive presence in the environment of many smart things, including sensors, actuators, embedded systems and other similar devices. Widespread connectivity, getting cheaper smart devices and a great demand for data, testify to that the IoT will continue to grow by leaps and bounds. The business models of various industries are being redesigned on basis of the IoT paradigm. But the successful deployment of the IoT is conditioned by the progress in solving many problems. These issues are as the following:

- The integration of heterogeneous sensors and systems with different technologies taking account environmental constraints, and data confidentiality levels;
- Big challenges on information management for the applications of IoT in different fields (trustworthiness, provenance, privacy);
- Security challenges related to co-existence and interconnection of many IoT networks;
- Challenges related to reliability and dependability, especially when the IoT becomes the mission critical component;
- Zero-configuration or other convenient approaches to simplify the deployment and configuration of IoT and self-healing of IoT networks;
- Knowledge discovery, especially semantic and syntactical discovering of the information from data provided by IoT.

The IoT technical session is seeking original, high quality research papers related to such topics. The session will also solicit papers about current implementation efforts, research results, as well as position statements from industry and academia regarding applications of IoT. The focus areas will be, but not limited to, the challenges on networking and information management, security and ensuring privacy, logistics, situation awareness, and medical care.

TOPICS

The IoT session is seeking original, high quality research papers related to following topics:

- Future communication technologies (Future Internet; Wireless Sensor Networks; Web-services, 5G, 4G, LTE, LTE-Advanced; WLAN, WPAN; Small cell Networks...) for IoT,
- Intelligent Internet Communication,
- IoT Standards,
- Networking Technologies for IoT,
- Protocols and Algorithms for IoT,
- Self-Organization and Self-Healing of IoT Networks,
- Object Naming, Security and Privacy in the IoT Environment,
- Security Issues of IoT,
- Integration of Heterogeneous Networks, Sensors and Systems,
- Context Modeling, Reasoning and Context-aware Computing,
- Fault-Tolerant Networking for Content Dissemination,
- IoT Architecture Design, Interoperability and Technologies,
- Data or Power Management for IoT,
- Fog—Cloud Interactions and Enabling Protocols,
- Reliability and Dependability of mission critical IoT,
- Unmanned-Aerial-Vehicles (UAV) Platforms, Swarms and Networking,
- Data Analytics for IoT,
- Artificial Intelligence and IoT,
- Applications of IoT (Healthcare, Military, Logistics, Supply Chains, Agriculture, ...),
- E-commerce and IoT.

The session will also solicit papers about current implementation efforts, research results, as well as position statements from industry and academia regarding applications of IoT. Focus areas will be, but not limited to above mentioned topics.

TECHNICAL SESSION CHAIRS

- **Cao, Ning**, College of Information Engineering, Qingdao Binhai University
- **Chudzikiewicz, Jan**, Military University of Technology, Poland
- **Zieliński, Zbigniew**, Military University of Technology, Poland

Wireless Agent-based Distributed Sensor Tuple Spaces using Bluetooth and IP Broadcasting

Stefan Bosse

University of Bremen, Dept. of
 Computer Science, and Institute
 for Digitization
 28359 Bremen, Germany
 Email: sbosse@uni-bremen.de

Abstract— Most Internet-of-Things (IoT) devices and smart sensors are connected via the Internet using IP communication directly accessed by a server that collect sensor information periodically or event-based. The spatial context (the environment in which the sensor or devices is situated) is not reflected accurately by Internet connectivity, and which is additionally not everywhere available. In this work, smart devices communicate connectionless and ad-hoc by using low-energy Bluetooth broadcasting available in any smartphone and in most embedded computers. Bi-directional connectionless communication is established via the advertisements and scanning modes. The communication nodes can exchange data via functional tuples using a tuple space service on each node. Tuple space access is performed by simple event-based agents. The Bluetooth Low Energy Tuple Space (BeeTS) service enables opportunistic, ad-hoc and loosely coupled device communication with a spatial context.

INTRODUCTION AND OVERVIEW

THE number of embedded systems grows exponentially. Ubiquitous and pervasive computing introduced visible and non-visible low-resource and mobile devices. Most Internet-of-Thing (IoT) devices and smart sensors are still connected via the Internet using IP communication that are accessed by a server that collect sensor information periodically or event-based. Internet access is not everywhere available. Additionally, the residential time of mobile devices can be short, not suitable for ad-hoc connection-based and negotiated communication. Finally, the spatial context (the environment in which the sensor or device is situated) is not considered (or inaccurately determined) by Internet connectivity. even new 5G technologies will not fully solve context-aware communication [1]. In this work, smart devices communicate connectionless and ad-hoc by using low-energy Bluetooth available in any Smartphone and in most embedded computers, e.g., the Raspberry PI or ESP32 devices. Bi-directional connectionless communication is established via the advertisement and scanning modes used in parallel. The communication nodes can exchange small data or functional tuples using a tuple space service. Mobile devices act as tuple carriers that can carry tuples between

different locations. Additionally, UDP-based Intranet communication can be used to connect tuple spaces.

Tuple spaces are widely used for data storage for loosely coupled distributed data processing systems. The Bluetooth Low Energy Tuple Space (BeeTS) service is a lazy distributed tuple space server and client. BeeTS uses Bluetooth and UDP broadcasting for tuple space interaction and tuple exchange. BeeTS supports tuples with an arity up to 4.

A tuple space provides a spatial context, i.e., tuple space access (by mobile devices) is limited to nearby devices, well suited for mobile networks [2]. Distributed tuple spaces can be connected by routers using IP communication if available. The router composes global space sets by tuple exchange and replication using hybrid rule- and event-based agents. These rules can be changed at run-time and the code can use Machine Learning algorithms to optimally distribute tuples under resource and spatial constraints.

The novelty of this work is two-folded. Firstly, an ubiquitous radio broadcast medium is used for low-distance communication in ad-hoc mobile networks combined with a unified tuple space paradigm. Secondly, the tuple space communication is performed by simple reactive event-based agents programmed in JavaScript that can be sent to a node via the tuple space operations, too. Agent-based message routing [3] is well suited in highly dynamic and unstructured network environments. The generative tuple space paradigm is well suited for ad-hoc mobile networks [4], especially if this paradigm is coupled with the agent paradigm [5].

COMMUNICATION

It is assumed that there is a broadcast medium B , e.g., using radio waves, which can reach a number of nodes $N_B = \{n_i\}_{i=1}^k$ defining a receive area/range coverage $C(B, N)(t)$ that changes over time t . B can send broadcast messages m to all listening nodes reachable by B . The set of nodes within B can vary on time and spatial scale. Furthermore, it is assumed that there is a probability $p_i(m, r_{i,j}, [t_0, t]) \in [0, 1]$ that a message m is received by a node i sent by node j in distance r within a time interval $[t_0, t]$. These two assumptions are fundamental for the proposed distributed tuple spaces.

It is assumed that single packets that can be send over B are strictly limited by a small number of bits, e.g., 200-300 Bits. This requires a compact and optimized message format, discussed in the next sub-section.

There are seven different message types:

- *OUT* stores a tuple in all tuple spaces receiving this message;
- *RD* and *INP* requesting tuples from all receiving tuple spaces;
- *TEST* checks for the existence of a tuple or set of tuples;
- *TUPLE* is either an initial message sending this tuple to all receiving nodes without; storing the tuple in the respective tuple space, or a reply to a tuple request;
- *IAMHERE* and *WHERE* messages are used for node search.

The message format consists of a message header and the data payload. The sequence number is required to detect the reception of multiple copies of the same message, a prerequisite for deployment with the Bluetooth device back-end that sends a message multiple times. The signature byte specifies the following tuple data pay-load. Depending on the back-end communication device and the supported packet format, the number of pay-load bytes can be very small. The signature field specifies the type of each tuple element with a tuple limit of four elements. For Bluetooth advertisement packages there is $N_{BLE}=32$, for the UDP back-end it is at most $N_{UDP}\geq 512$. The message header and the data payload is encoded in an BLE advertisement packet using one device local name attribute (ASCII85 encoded) and seven 16 Bit service UUID attribute fields.

In contrast to typical tuple space services, the tuple operations are not atomic here. They can be executed at any given time point t in the near future or never, and the set of reachable tuple spaces that execute the request is not bound and can be zero. There is no assumption that neither a message arrives at a specific node nor that request is processed successfully. There are filter rules processed by agents that can be prevent tuple operation execution, too. That means, the *INP* operation is only a suggestion to all receiving tuple spaces to remove a matching tuple. All operations pose a probabilistic behaviour, i.e., there is a probability ≥ 0 that a message is processed.

The encoding of tuples is done automatically. Before a tuple is encoded and packed, a signature is derived, numbers are classified either in integer 16 Bit or float 32 Bit values depending on the actual value.

Devices can access remote tuple spaces of nearby neighbouring nodes (typically in the range of 1-10m) by using BLE broadcasting (called ble-ts). A device in advertisement mode will send out periodically advertisement message that contain a small payload depending on the advertisement message class. In this work, the pay-load is

limited to 32 Bytes. There are 40 RF channels in BLE, each separated by 2 MHz (centre-to-centre). Three of these channels are called the primary advertising channels (labelled 37, 38, and 39), while the remaining 37 channels are called the secondary advertisement channels (they are also the ones used for data transfer during a connection). The primary channels are switched randomly in periods. On the other side, the scanning devices has to switch the (primary) receiving channels randomly, too. There is a probability p , that an advertisement packet is received if both scanner and advertiser are switch on the same channel and if there is no other sending within the receiving range creating collision (invalidation of the message).

In addition to the BLE broadcast communication, nodes that are connected to a local IP network can exchange tuple requests via UDP broadcast messages (called udp-ts). Although, UDP messaging is theoretically reliable, UDP broadcasting via wireless connections is not supported. Security is provided by a symmetric two-way encryption with format-preserving encryption of tuple messages using byte look-up tables.

BEETS

The principle network architecture combining Bluetooth and UDP-IP broadcast communication technologies is shown in Fig. 1. Tuple messages can either be sent via Bluetooth advertisement (based on [7]) or via single UDP packets within a local IP network. BeeTS is programmed entirely in JavaScript and can be executed by node.js with a Bluetooth socket modules for BLE access, the *noble* module for the central BLE part, and *bleno* for the peripheral part. Note that BeeTS uses the peripheral and central (master) mode simultaneously (advertising and scanning), requiring a Bluetooth device with version ≥ 4.0 . BeeTS is basically a small library module written in JavaScript. Smartphones act as mobile devices and provide both a rich set of sensors and BLE connectivity. Each communication back-end can receive tuple requests. If there is a listener installed for tuples (with pattern matching), incoming tuples (*TUPLE* message) can be consumed by the listener or not. Otherwise, incoming tuples are stored in the local tuple space.

There are agents acting as a bridge between the communication back-end and the tuple space. They can filter incoming messages and decide to reply immediately, to access the tuple space, or to discard the message. Agents are functional code that listen to incoming tuple requests. There can be more than one agent. Communication between agents is established via the tuple space, too.

A broadcast message sending via BLE enables the advertisement mode of the device for a specific time interval $[ts,te]$, shown in Fig. 1 (a). The duration of the time interval Δt determines the receiving probability, the collision probability (if more than one station is sending), the number of advertisement packets that contain the message m , and the number of different messages that can be sent per second. The interval time Δt must at least $3 \times t_{sw}$, with t_{sw} as the

average channel switching time of the sender (and receiver). It is assumed that the sender and receiver have the same switching time, typically 100 ms. Important to note that channel switching introduces small dead time intervals (about 1-10ms). A suitable value for Δt is about 500ms.

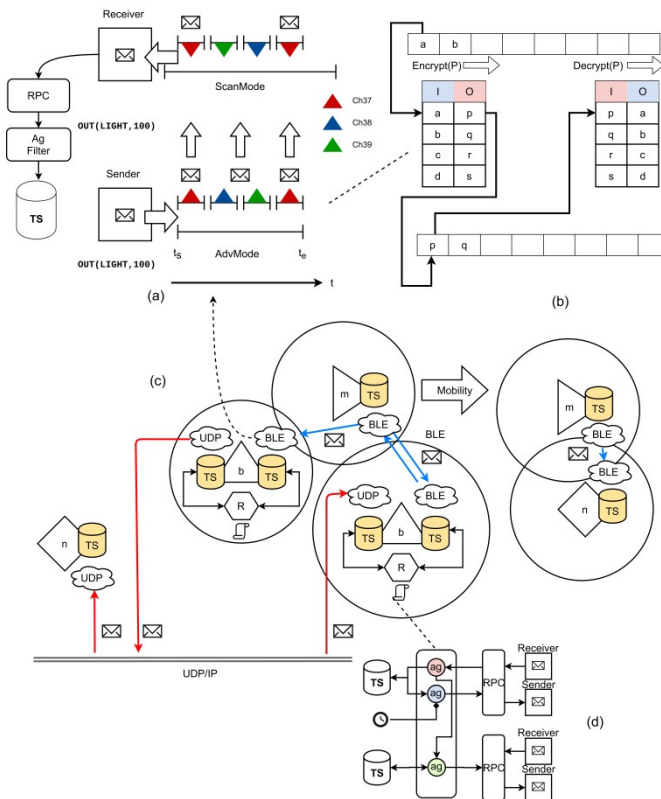


Fig. 1. (a) BLE communication (b) Security by FPE (c) The hybrid network architecture using BLE and UDP-IP broadcast communication; n: stationary node, b: Stationary beacon, m: Mobile node, TS: Tuple space, R: Rule-based router (d) Message routing by agents between different TS

Each physical communication interface (BLE/UDP) is attached to its own tuple space, i.e., there are two distributed space sets connected via BLE and UDP, respectively. This division is grounded in the spatial context of tuple spaces. Using BLE communication only nearby nodes can insert or remove tuples, whereas UDP communication enables tuple exchange over short and large distances, too. Tuple exchange between BLE and UDP tuple spaces is provided by a customisable router, shown in Fig. 1 (c). Application-specific routing rules (functional code) provide transfers based on patterns and content of tuples. The rule set is dynamic and can be changed at run-time. The router extends the visibility and scope of tuples based on adaptive code. The code can use Machine Learning, e.g., reinforcement learning, to improve tuple space distribution. The routers connect local spaces and compose organised global spaces.

Each time a message is received it is passed to the Remote Procedure Call layer (RPC). Among the message data, the sender MAC ID, a time stamp, and the signal strength is added to the message.

The BeeTS framework basically provides a communication platform using radio communication like Bluetooth or WLAN. The communication bandwidth of various devices can be significantly limited (e.g., in the case of Bluetooth advertisement mode that can be only 2 messages/second). One main feature of BeeTS nodes is the capability to execute event-based reactive agents programmed in JavaScript that perform, e.g., filtering of incoming tuple space requests. An agent is functional data consisting of private body variables (including functional values) and event handlers that are activated by incoming messages, sensors (if the host platform provides them), periodically, or only one time. Agents are executed in sandboxed containers and are used for message filtering and routing between different tuple spaces, shown in Fig. 1 (d). The format-preserving encryption of tuple messages using look-up tables is shown in Fig. 1 (b).

USE-CASE: DISTRIBUTED SMART BUILDING CONTROL

This use-case deploys three different node classes implementing a distributed building light control system:

1. Stationary beacons (Raspberry 3) equipped with BLE and WLAN connectivity and supporting ble- and udp-connected tuple spaces in both test and production deployment;
2. Mobile devices (battery powered RP Zero stacked with a smartphone for testing, stand-alone smartphone in production systems) supporting ble- and udp-connected tuple spaces only in production environments;
3. A central monitoring and light control service supporting udp-connected tuple spaces.

Each node deploys at least one event-based agent that implements necessary node operations like interaction with mobile devices or users, and tuple filtering and bridging. Beacons consume and aggregate mainly sensor data from mobile (sensorised) devices like smartphones and forward micro-surveys from the central server to mobile devices. But beacons can initiate and manage micro-surveys, too. To minimise the number of sent tuples via the BLE device, the mobile nodes monitor the user behaviour by analysing the accelerometer and gyroscope sensors. Updates of light sensor tuples are only sent if either the light conditions changes or the mobile device was moved in space. For rapid prototyping, smartphones are using generic Web browser loading an application page from the locally attached Raspberry PI zero bundled with the smartphone. All sensor data is sent to the embedded computer that executes the mobile application logic and that performed the BLE communication.

Mobile devices use their light sensor in conjunction with accelerometer and gyroscope sensors to estimate the ambient

light conditions and the user mobility by classifying the user activity in rest, smartphone use, and movement phases.

The measured light sensor data is processed by a sensor agent that tries to estimate if the smartphone is currently exposed to external light or if it is stored in a box. If external light is detected, sensor light tuples are sent via BLE. Nearby beacons distributed in the building about every 10-20m (and one per room/floor) collect these tuples and send aggregated sensor light values to the central server via udp-connected tuple spaces.

Among sensor tuples, there are micro-survey tuples that are sent from a beacon (initially delivered by the central server via the UDP tuple space) to mobile devices. If a device supports HMI (e.g., a smartphone), a short question is posted to a chat dialogue platform embedded either. The user can answer the question and the answer is passed back to the beacon (or any other beacon due to movement). The beacons collect the micro-survey replies and forward them to the central server.

For the evaluation of the loss rate of BLE tuple space communication, a partial set-up was chosen with four beacons at four different spatial positions and four mobile devices here all at the same position. An average loss below 10% can be achieved within a radius of about 5m. Some nodes can communicate over larger distances up to 10m. The tuple message send time interval has no significant impact on the loss rate within time interval [500s,2000s] and with this (small) set-up. If the number of nodes within the radio range increases, the loss rate will increase.

CONCLUSION

In this work, The Bluetooth Low Energy Tuple Space (BeeTS) service enables opportunistic, ad-hoc and loosely coupled device communication with distributed tuple spaces that are used to exchange data between devices providing a spatial context with respect to data and communication. Smart devices access the tuple spaces by tuple message communication using event-based agents. The communication is connectionless and ad-hoc by using low-energy Bluetooth broadcasting available in any Smartphone and in most embedded computers, e.g., the Raspberry PI devices. Bi-directional connectionless communication is established via the advertisement and scanning modes used in parallel by transferring encoded tuple messages. Mobile devices act as tuple carriers that can carry tuples between different locations. Additionally, UDP-based Intranet communication can be used to connect tuple spaces. with spatial context. Multiple independent tuple spaces can be serviced on one network node bridged by agents. Among the tuple spaces, BeeTS implements simple reactive event-based agents. These agents perform tuple space management, interaction between devices and users, and they act as tuple filters and forwarders between multiple tuple spaces. A preliminary study showed the suitability of the broadcast communication for distributed ad-hoc networks preserving a spatial context lacking in other approaches. Analysis showed

a low loss of BLE broadcast messages (about 10-20%) but higher and unpredictable loss rates of UDP broadcast communication using WLAN, even if N:1 unicast communication was used to simulate broadcast messages.

APPENDIX: EXAMPLE AGENT

The following example shows an event-based agent programmed in JavaScript reacting on tuple messages. It consists of body variables and the event section. A specific tuple space can be selected.

```
var agent = {
  x : 0,
  y : 0,
  ..
  on : {
    'ts.udp:(TIME,?):' : function (ev) {
      ts.ble.notify(ev.tuple);
      return consumed?;
    },
    'ts.ble:(SENSOR,?,?,?):' : function (ev) {
      log(ev.tuple, ev.from, ev.rssi)
      ts.udp.out(['EVENT',
        JSON.stringify(ev.tuple),
        ev.from, ev.time]);
      return consumed?;
    },
    init : function () {
      this.x=random(1,1000);
    },
    1000 : function (counter) {
      // called each 1000 ms
      return true
    },
    'sensor.light:abs(sensor-sensor0)>50' :
      function (ev) {
        if (ev.sensor>500)
          ts.ble.notify(['ALARM',
            'LIGHT', 'HIGH']);
      }
  }
},
Agent.create(agent);
```

REFERENCES

- [1] A. Gupta and R. K. Jha, "A Survey of 5G Network: Architecture and Emerging Technologies", IEEE Access, vol. 3, 2016.
- [2] P. Costa, L. Mottola, A. L. Murphy, G. P. Picco. "Teenylime: transiently shared tuple space middleware for wireless sensor networks." In Proceedings of the international workshop on Middleware for sensor networks, pp. 43-48. 2006
- [3] E. Shakshuki, H. Malik, and X. Xing, "Agent-Based Routing for Wireless Sensor Network", Lecture Notes in Computer Science, Advanced Intelligent Computing Theories and Applications. With Aspects of Theoretical and Methodological Issues, vol. 4681, pp. 68-79, 2007.
- [4] N. Davies, A. Friday, S. P. Wade, G. S. Blair, "L2imbo: A distributed systems platform for mobile computing", Mobile Networks and Applications 3(2), 143-156., 1998
- [5] S. Bosse, "Unified Distributed Sensor and Environmental Information Processing with Multi-Agent Systems: Models, Platforms, and Technological Aspects", ISBN 9783746752228, epubli, 2018
- [6] S. Bosse, "BeeTS: Smart Distributed Sensor Tuple Spaces combined with Agents using Bluetooth and IP Broadcasting", CoRR abs/2204.02464, 2022
- [7] M. Nikodem and M. Bawiec, "Experimental Evaluation of Advertisement-Based Bluetooth Low Energy Communication", Sensors, vol. 20, no. 107, 2020

Resource Partitioning in Phoenix-RTOS for Critical and Noncritical Software for UAV systems

Hubert Buczyński, Krzysztof Cabaj
Warsaw University of Technology
Nowowiejska 15/9, 00-665 Warszawa, Poland
Email: {hubert.buczynski.dokt, krzysztof.cabaj}@pw.edu.pl

Paweł Pisarczyk
Phoenix Systems Sp. z o.o.
Ostrobramska 86, 04-163 Warszawa, Poland
Email: pawel.pisarczyk@phoenix-rtos.com

Abstract—Modern embedded systems’ increasing complexity and varied safety levels make it hard to coordinate all functionalities within a single run-time environment. Access to more advanced and capacious hardware changes the trend from utilising many separated platforms into one managing the whole compounded airborne system. Providing an appropriate isolation and synchronisation level is achievable only by adapting an operating system with separation mechanisms into UAV systems. This paper introduces Phoenix-RTOS, the microkernel structured real-time operating system designed to be consistent with aerospace standards DO-178C and ARINC 653. The current market offers many counterparts like VxWorks, Integrity 178, PikeOS and many others. These products are well known and used in leading-edge avionics and space projects. However, it is not possible to use them in many low-budget projects due to the high price. The Phoenix-RTOS differs from others and is an open-source project becoming a standard solution for energy, gas meters and UAV systems.

In this paper, we focus on the currently designed mechanisms of microkernel architecture for providing a mixed-criticality system, particularly for compliance with ARINC 653. Engineers have been identifying time and space partitioning issues to cope with tight worst-case execution bounds of critical tasks.

I. INTRODUCTION

THE Unmanned Air Vehicles (UAV) market increased significantly in recent years. However, plans about flying machines carrying out missions of particular importance (e.g. transport of hazardous substances and medical supplies) above our heads in inhabited areas cannot become a reality without ensuring the high safety standards of airborne systems. According to European Union regulations introduced in 2019 [1], the UAV can be allowed to conduct a mission in the urban environment only after a positive certification process conducted by the civil aviation agencies such as European Union Aviation Safety Agency (EASA) or Federal Aviation Agency (FAA). To reduce the time of the certification process, the manufacturers choose certified Real-Time Operating Systems (RTOS) to run their critical and non-critical applications. The current market offers several operating systems compliant with aviation standards, but only large companies can afford them due to high license costs. To overcome this obstacle Phoenix Systems has started working on Phoenix-RTOS 178 version compliant with aviation standards [1]. Making a system as an open source product allows manufacturers with a limited budget to enter the market. The new version of the Phoenix-

RTOS is designed to comply with DO-178C and ARINC 653 standards described in the next paragraph.

A. Aviation Standards

Standards play a crucial role in the aerospace industry. One of the most important is DO-178C, the fulfilment of which allows usage software in airborne devices. DO-178C defines five levels of failure caused by software: catastrophic, hazardous, major, minor, and no effect. Moreover, it defines five corresponding Design Assurance Levels (DAL) from the most rigorous level A to level E. Depending on the DAL level, tenants’ appropriate process management has to be fulfilled. Another one, Aeronautical Radio Incorporated (ARINC) standard focuses on technical aspects that define a series of detailed rules for a dedicated area, like data transmission protocols, cockpit user interfaces, and many others. The aerospace projects are always compliant with DO-178C, however meeting the ARINC standards depends on project’s requirements. The best policy for operating system development for critical applications is described in standard ARINC 653. Running different critically level software on the same hardware platform is the primary domain of Integrated Modular Avionics (IMA). To realize an IMA approach in RTOS, the Partitioning Operating System (POS) concept should be introduced to support spatial and time partitioning [10]. In the avionics industry, the Avionics Application Standard Software Interface, called APEX provided by ARINC 653, has been introduced as a set of guidelines to provide a standardized interface between POS and avionics applications [10]. The leading role of the ARINC 653 is to improve the safety and certification process of safety-critical systems and outline the architectures approach for POS’s engineers [10]. More details about APEX are presented in the fourth section.

B. History of Phoenix-RTOS

The idea for writing a new RTOS originates from the Warsaw University of Technology where the prototype of Phoenix-RTOS was developed from scratch as a master thesis [1] in 1998. The rapidly growing Internet of Things (IoT) market resulted in an industry need where a new operating system with efficient implementation and rich functionalities to be required.

Due to this fact, the second version of Phoenix-RTOS was developed by Phoenix Systems and widely implemented in data concentrators for the smart grid, smart energy meters and smart gas meters. Phoenix-RTOS 2 is recognized on the market as a real-time system for the smart grid and software-defined solutions.

The third version of the system based on a microkernel architecture has been developed to be easily used in microcontroller-based low power devices and more advanced processors. The new approach provided in version three allows the use of Phoenix-RTOS on a broad range of processors architecture from the smallest ones like ARM Cortex-M, to more those complex like ARM Cortex-A, ia32 or RISC-V [1]. Employing the Phoenix-RTOS in version three to the energy meters sector has proven its applicability for high complexity systems which work in harsh environments. The experience obtained in the industry moved the company forward in achieving the next milestone to make Phoenix-RTOS consistent with DO-178C standard in a design assurance level A.

Phoenix-RTOS compliant with DO-178C is developed as an open-source project under the BSD license. From the time of writing this paper, only several open-source competitors supporting ARINC 653 have been found, which are described in the third chapter. The rest of the systems for critical purposes are commercial. It is strongly believed that by providing the system as an open-source product, the gap in the market is filled for projects with a limited budget, and the topic of critical systems will be covered.

This paper presents our consideration for partitioned based microkernel development in Phoenix-RTOS. A current trends analysis in the industry has been conducted and there are visible alternatives that can be offered by the open-source product. The research focuses on implementing microkernel mechanisms to support spatial and time partitioning to fill ARINC 653 requirements.

II. MIXED CRITICALITY SYSTEM

Operating systems for critical purposes should be highly reliable. As is considered in *Tanenbaum et al.* [5] they should not be interrupted entirely or halted to recover from a failure that occurs in a module that is not in the critical execution path of a service or an internal operation [4]. To achieve this goal, operating systems should provide safety and security functionalities not to allow the non-critical zone's fault to propagate to a critical one. Safety and security functions seem to be similar, and although these terms have common elements, they differ. This chapter presents the concept of safety-critical and security-critical systems in more detail. The next one presents examples of systems divided into appropriate categories.

A. Safety-critical

The main challenge tackled by a safety-critical system is to provide a clear border between different critical zones called partitions. Due to this fact, these kinds of operating systems

are often called partitions kernels. Partitioning mechanisms should provide spatial and temporal separation [2]. Moreover, the partition kernel is in charge of resources management such as I/O devices to permit access to only assigned parts of the system. To enforce spatial separation, each individual partition runs in a hardware-protected address space. For this purpose, the Memory Management Unit (MMU) performs mapping of virtual to physical addresses. To enforce time separation, each partition and thread have a dedicated time slot of the CPU in which the actions have to be completed. The static analysis is obligatory to define the Worst-Case Execution Time (WCET) of each running partition and process. Based on the WCET the best scheduling algorithm is selected to take control of the CPU when attempts of time overrun occur. The central concept of a safety system requires static resource allocation for partitions and static analysis of system performance. Safety-critical systems are compliant with DO-178C and ARINC 653. However, standards do not impose requirements describing how spatial and time separation should be performed. The chosen solutions can differ between individual systems.

B. Security-critical

The security-critical systems originate from the concept of Multiple Independent Levels of Security (MILS) specification, which is a high-assurance architecture based on separation and information flow security [3]. The foundation of MILS is a Separation Kernel (SK), which is responsible for adherence of data isolation, damage limitation and resource partitioning. SK extends partitioning kernels of sets of specific functionalities to enforce security separation, and information flow [2]. The security requirements are known as Common Criteria (CC) [3] establishing seven Evaluation Assurance Levels (EAL) from 1 to 7, which is the most rigorous. Moreover, CC defines robust requirements dedicated to an operating system called Separation Kernel Protection Profile (SKPP). To obtain satisfying certification for EAL7, the partitioning mechanisms have to be rigorously verified using formal methods [3]. Compared to partitioning kernels, SK provides a more dynamic environment that does not have to be known ahead of time. The typical approach to realizing separation kernel is based on embedded hypervisor and software virtualization mechanisms. The hypervisor layer manages the system resources and virtual partitions, which are capable of running guest operating systems with different levels of critically [2].

III. BACKGROUND AND RELATED WORK

Operating systems for critical applications is a complex topic that demands a considerable workloads and financial resources. Due to this fact, only several companies can deliver reliable products. The most known and widely used systems on the market are presented in the following subsections. The last paragraph is devoted to open source projects.

A. Lynx Software Technologies

The Lynx Software Technologies has on offer two operating systems for critical application: LynxOS-178 and LynxSecure.

The first version of the system, LynxOS-178, is a safety-critical system. It meets the DO-178C DAL A regulations and supports POSIX, and APEX standards [6]. However, this version of OS would not meet the highest EAL7 criteria, which require formal mathematical methods to prove the security of SK. For this reason, LynxSecure is introduced to provide a separation kernel based on hypervisor virtualization. The second version OS design includes safety and security domain isolation, trusted execution environments and reference monitor plugins such as firewalls and encryption [6]. A popular solution for complex systems is to use LynxSecure to provide secure partitioning and LynxOS-178 to launch critical applications compliant with APEX.

B. GreenHills

INTEGRITY 178 is a world-leading RTOS for safety and security applications certified both to the highest level of DO-178C DAL A and SKPP/EAL6+ [7]. The producer claims tasks protection in the guaranteed time window domain. The space domain is arranged by static memory allocation and hardware enforcement of MMU for each partition. The system isolates applications and data into different security domains to ensure the MILS environment. As a separation kernel, INTEGRITY 178 provides a virtualization layer in the userspace instead of in the kernel [7].

The GreenHills also offers a version for multicore architectures - INTEGRITY 178 tuMP. As one of the few systems that provide full flexibility in choosing the software multi-processing architecture, ranging from simple Asymmetric Multi-Processing (AMP) to modern Symmetric Multi-Processing (SMP) to Bound Multi-Processing (BMP) for the highest combination of determinism and utilization [7].

C. WindRiver

The WindRiver company takes a similar approach as Lynx Software Technologies. In the offer can be found two versions of software for critical purposes. The first one VxWorks 653 is compliant with DO-178B/C DAL A and ARINC 653 [8]. The second one VxWorks MILS provides compliance to SKPP using a hypervisor. To combine safety and security mechanisms, two versions of WindRiver products are used side by side.

D. SYSGO

The flag product of SYSGO company is a PikeOS compliant with DO-178C DAL B and ARINC 653 [2]. Although the system provides safety and security-critical mechanism, it is not compliant with SKPP. PikeOS offers a separation kernel-based hypervisor with multiple partitions for many other operating systems and applications [9]. The engineers from SYSGO developed a similar concept as the GreenHills company for INTEGRITY-178. The PikeOS does not need an external hypervisor, its internal layer provides security and virtualization between partitions.

E. Open Source Systems

According to a survey presented by researchers in [2], the majority of open-source projects are academic implementations. Special consideration should be given to POK and XtratuM operating systems due to compliance with ARINC 653. However, none of them is compliant with DO-178C. Furthermore, there are other systems worth mentioning like Quest-V or Muen. Although they provide some mechanism of spatial and temporal partitioning of resources, they do not comply with any of the considered standards.

IV. ARINC 653

The primary objective of ARINC 653 is to fulfil the IMA requirements to provide an environment for the independent execution of avionics applications utilizing different critical levels. The majority scope of the regulation defines the Application Programming Interface (API) between the operating systems and avionics applications. One of the benefits of standard usage is to provide high portability of avionics software to run on different operating systems.

The standard consists of five parts [11]. Part 0 describes a general overview about ARINC 653. Part 1 summarises the required services to be supported by APEX and part 2 reports extended services. Section B in chapter fourth, precisely presents the services concepts. Part 3 of ARINC 653 is divided into two parts: 3A and 3B, and provides a guideline for conformity tests for both required and extended services. Part 4 defines a subset of API specified by part 1. Finally, part 5 describes recommended capabilities for multicore applications. The conformance of the APEX API to the ARINC 653 standard is validated by passing all of test suites defined by part 3 of the standard.

A. General architecture

Fig. 1 shows the example architecture of a system compliant with ARINC 653. The integrated module represents a complete RTOS which consists of a core module and applications. The Core Software (CSW), referred to as the operating system and hardware layer applying many processors, each consisting of one or more processor cores [11]. CSW should provide the ARINC 653 interface, health monitoring system, two level scheduler and suitable time and space separation based on configuration data. The highest layer of the IMA architecture presents either application software partitions and system partitions. The first ones are restricted to using only ARINC 653 calls to interface to the system [11]. The second ones are partitions specific for the operating system requiring interfaces outside of the APEX scope. They manages system specific task such as device drivers or fault management schemes [11] which are not defined by ARINC 653. Both of them are subject to robust space and time partitioning.

B. Static System Configuration

According to the specification, a partition is a program in an application environment that consists of text, data section, stack, own context, and configuration attributes [11]. In this

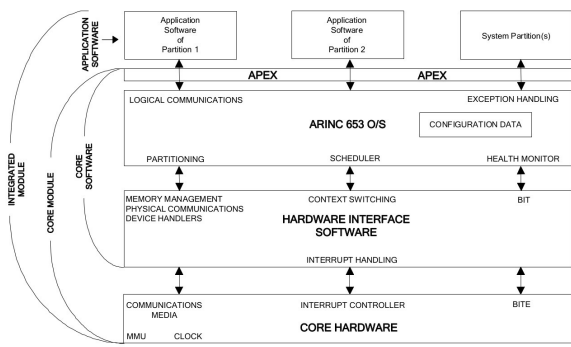


Fig. 1. Example of Integrated Module Architecture [11].

case, a process is a programming unit contained within a partition that executes concurrently with other processes within the same partition [11]. In other words, the process in ARINC 653 glossary corresponds to a thread running inside a process, and the partition refers to a process in UNIX like systems.

Each of the partitions and processes are described by system configuration files. They contain detailed information about the demand for system resources, communication ports, execution windows and scenarios for health monitor systems. The most popular approach for APEX environment description is an Extensible Markup Language (XML) or AADL language.

V. PHOENIX-RTOS 178

Phoenix-RTOS 178 is a new version of the system compliant with DO-178C and ARINC 653, based on the Phoenix-RTOS version three. The new approach follows the rules for mixed safety-critical systems to provide a reliable and robust run-time environment for the avionics industry.

This chapter focuses on the system architecture to explain how spatial and time separation is resolved in Phoenix-RTOS 178. The presented solutions have been developed and deployed on the Zedboard platform, equipped with a Xilinx Zynq-7000 dual-core ARM Cortex-A9 processor family.

A. System Architecture

The system architecture consists of a bootloader and a kernel. The first one, Phoenix-RTOS loader (PLO) can be treated as a first-stage and a second-stage bootloader. Acting as the first-stage bootloader, PLO configures the memory controllers and various supported devices on a dedicated platform. It is also responsible for preparing the board for the kernel and performing built-in tests (BITs) to check correctness of the operation of the critical peripherals. The second-stage bootloader loads the operating system and selected partitions from storage devices to the memory. The fig. 2 shows PLO Command-Line Interface (CLI). Configuration data about the system setup is passed to the kernel by a structure called syspage. The syspage contains all information about board configuration and partition descriptions compliant with ARINC 653 standard. The second system's component is a kernel responsible for providing spatial and time separation by memory and threads

```
Phoenix-RTOS loader v. 1.21 rev: 253ed19
hal: Cortex-A9 Zynq 7000
dev/uart: Initializing uart(0.0)
dev/uart: Initializing uart(0.1)
dev/usb: Initializing usb-cdc(1.2)
dev/flash: Initializing flash(2.0)
cmd: Executing pre-init script
console: Setting console to 0.1
Waiting for input, 200 [ms]
(plo)%
```

Fig. 2. Phoenix-RTOS loader running on Zynq-7000.

```
Phoenix-RTOS microkernel v. 2.97 rev: aa8e57e
hal: Xilinx Zynq-7000 ARMv7 Cortex-A9 r3p0
hal: ThumbEE, Jazelle, Thumb, ARM, Security
hal: Using gic interrupt controller
vm: Initializing page allocator (876+0)/131072KB, page_t=16
vm: [256x][24K][6P][H][17K][36A][127H]PPPP[805.]PPPS[31744.]
vm: Initializing memory mapper: (8105*64) 518720
vm: Initializing kernel memory allocator: (64*48) 3072
vm: Initializing memory objects
proc: Initializing thread scheduler, priorities=8
syscalls: Initializing syscall table [101]
main: Starting syspage programs: 'dummyfs', 'zynq7000-uart', 'psh'
(psh)%
```

Fig. 3. Phoenix-RTOS shell running on Zynq-7000.

management modules. The system is startup by the initial thread being in charge of launching the health monitor, system and user partitions with resources assigned to them by the setup data. After the successful initialization phase, the chosen scheduler algorithm is launched. The fig. 3 shows Phoenix-RTOS providing interaction with user via Phoenix-RTOS shell. The essential kernel's safety-critical mechanisms are described in the next two chapters.

B. Spatial Separation

The crucial element of the system for critical purposes is providing spatial separation for partitions. In Phoenix-RTOS 178, this mechanism is implemented using multi maps and an MMU controller. The multi maps approach allows the simultaneous use of different memory devices such as on-chip RAM (OCRAM) or DDR SDRAM. This solution provides additional physical separation. User is able to use a specific memory for dedicated purposes. If there is a requirement to use the memory with a fast access for critical application, the OCRAM should be chosen. The memory virtualization is provided by the MMU controller and configured based on the information about accessible volatile memory described by the PLO in the syspage.

Another essential element embedded into the memory management module is a caching policy. Phoenix-RTOS 178 invalidates and cleans executable pages and allows the user to define uncacheable memory regions independently.

The memory virtualization provides the same linear memory map for each partition. Switching between them provides an additional delay in changing the MMU controller's translation table. To avoid an overhead, the Address Space Identifier (ASID) mechanism is added to the memory management module to assign a memory map to the partition. Furthermore, the spatial separation is in charge of controlling of the mapping

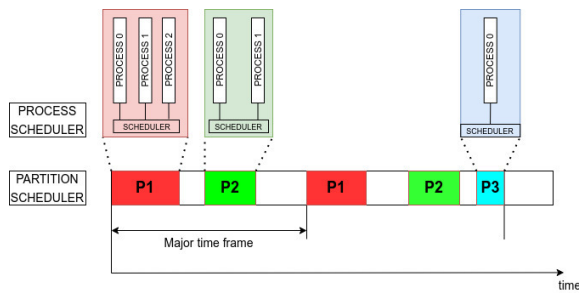


Fig. 4. Example of partition and processes scheduling.

I/O devices' memory to only one partition to not allowing sharing resources.

C. Time Separation

A robust and reliable scheduling policy is an essential element of the operating system to perform critical tasks on time. In an environment compliant with ARINC 653, a two-level scheduler should be introduced. The fig. 4 shows an illustrative scheduling policy for mixed-criticality systems.

The first scheduler supervises the partition work. The most common algorithm for hard real time embedded systems is the Rate Monotonic (RM) scheduling with a fixed priority [12]. In the presented example, each partition has: an assigned priority, WCET, Average Case Execution Time (ACET) and period. The critical partitions have the highest priority allowing to preempt the other ones and to meet deadlines. Making a static analysis using the integration tool, it is possible to verify whether the system is feasible and all partitions can meet their deadlines with the assumed values.

The second scheduler is responsible for scheduling processes within a partition. The ARINC 653 allows to choose a dedicated scheduling policy or the default one is selected.

Keeping WCET bounds of partitions depend on the internal mechanism of the operating system. The common problem for RM scheduling policy is a priority inversion, causing deadlock between partitions or processes. To avoid this situation, the synchronization mechanisms owning dynamic changing priority to the highest one sharing amongst other partitions or threads.

D. APEX Interface

Realization of the APEX services in Phoenix-RTOS 178 is performed using systems calls to the kernel. The interface is implemented in the form of a static library linked to the partition executive file. The library interface checks the correctness of the given arguments and passes them via the Application Binary Interface (ABI) to the kernel running in a privileged mode.

The microkernel architecture provides a message passing interface between servers. The communication services like

inter-partition communication use the exact mechanism. The other ones use a dedicated system calls to perform the action defined in ARINC 653.

VI. CONCLUSION

In this paper, we presented the research work on the mechanisms for integrating critical and noncritical software management by safety operating systems. The growing UAV market and its application will need to use reliable RTOS compliant with DO-178C and ARINC 653. Phoenix-RTOS 178, as an open source project, will be the best choice for manufacturers entering the market with a limited budget.

The presented work shows working Phoenix-RTOS on the Zynq-7000 platform commonly used in airborne systems. Our goal was to highlight the basic concepts of mixed criticality systems and explain the general architecture of Phoenix-RTOS 178. Future work includes completing time and space separation mechanisms as well as the APEX interface in Phoenix-RTOS 178.

VII. ACKNOWLEDGEMENTS

The presented concepts are the part of the project - *Phoenix-RTOS 178 - new version of the Phoenix-RTOS operating system dedicated for implementation of systems compliant with the guidelines of the DO-178C safety standard* developed by the Phoenix Systems Sp. z o.o.. The project POIR.01.01.01-00-1412/19-00 is co-financed as a grant agreement by European Union represented by the National Centre for Research and Development (NCBiR) in Poland.

REFERENCES

- [1] Phoenix Systems Sp. z o.o. - Homepage, <https://phoenix-rtos.com/documentation>. Last accessed 4.12.2021
- [2] Y. Zhao, D. Sanan, F. Zhang, and Y. Liu, "High-Assurance Separation Kernels: A Survey on Formal Methods." arXiv, 2017. doi: <https://doi.org/10.48550/arXiv.1701.01535>.
- [3] J. A. Foss, P. W. Oman, C. Taylor, and W. S. Harrison, "The MILS architecture for high-assurance embedded systems," *International Journal of Embedded Systems*, vol. 2, no. 3/4. Inderscience Publishers, p. 239, 2006. doi: <http://dx.doi.org/10.1504/IJES.2006.014859>.
- [4] M. Ahmed and S. Gokhale, "Reliable Operating Systems: Overview and Techniques," *IETE Technical Review*, vol. 26, no. 6. Medknow, p. 461, 2009. doi: <http://dx.doi.org/10.4103/0256-4602.57831>
- [5] A. S. Tanenbaum, J. N. Herder, and H. Bos, "Can We Make Operating Systems Reliable and Secure?," *Computer*, vol. 39, no. 5. Institute of Electrical and Electronics Engineers (IEEE), pp. 44–51, May 2006. doi: <https://doi.org/10.1109/MC.2006.156>.
- [6] Lynxks - Homepage, <http://www.linuxworks.com/>. Accessed 4.12.2021
- [7] GreenHills - Homepage, <https://www.ghs.com/>. Accessed 4.12.2021
- [8] WindRiver - Homepage, <https://www.windriver.com/products/vxworks>. Accessed 4.12.2021
- [9] PikeOS Homepage, <https://www.sysgo.com/pikeos>. Accessed 17.12.2021
- [10] Delange J and Lec L. Pok, An ARINC653-compliant operating system released under the BSD license. In: Proc. of the 13th Real-Time Linux Workshop, Prague (Czech Republic) 2011
- [11] Aeronautical Radio, Inc., 2010, Avionics Application Software Standard Interface: ARINC Specification 653P0-1, 653P1-3
- [12] S. Siewert and J. Pratt. Real-Time Embedded Components and Systems with LINUX and RTOS. ISBN: 978-1-942270-04-1

Data Exchange Protocol for Cryptographic Key Distribution System Using MQTT Service

Janusz Furtak

Military University of Technology
 ul. Kaliskiego 2, 00-904 Warsaw,
 Poland
 Email: janusz.furtak@wat.edu.pl

□ **Abstract**— There is an increasing demand for capturing reliable data from IoT network devices. Due to the limited capabilities of such devices to process and store sensitive data and the range and performance of the communication link, it is a significant challenge to develop a secure solution for symmetric key distribution. This paper presents a secure data exchange protocol for a cryptographic key generation, renewal, and distribution (KGR) system. The Trusted Platform Module (TPM) supports the trust establishment, key generation, all cryptographic procedures of the KGR system and secure data exchange in the described protocol. The protocol uses the MQTT (Message Queuing Telemetry Transport) service, which IoT devices widely use.

I. INTRODUCTION

IN many systems where symmetric cryptography has been decided upon, the challenge of securely generating, renewing and distributing symmetric cryptographic keys must be met. For traditional IT systems, solutions of this type are familiar. A more significant challenge is to develop a solution for scenarios where IoT sensor nodes are the data source. The reason is that sensor nodes are usually built on devices with limited capabilities. These limitations include memory size, processing power, limited power source, and the short range of the wireless link used. Using a certification authority (CA) to build trust structures in such systems is difficult or impossible. Hence, in the cryptographic key distribution system and in systems that receive those keys, trust structures must be created locally with the support of electronic circuits such as TPM or Zymkey¹.

Large networks of sensor nodes are usually divided into groups of cooperating devices to find a compromise between the limited capabilities of IoT devices and security requirements (e.g., confidentiality, integrity, availability, etc.). This group of devices uses group key management (GKM) for key distribution. Dammak et al. in [1] presented an extensive analysis of the properties of GKM systems

depending on their applications. The analysis included: wireless body area networks (WBANs) [2], wireless sensor networks (WSNs) [3], cloud computing [4], wireless IPv6 networks [5], and IoT [6][7]. The result of this analysis is the following conclusions. Performance degrades quickly when many devices form a single group and when group members change frequently. The problem of key renewal when devices join/leave a group is not fully resolved. The Decentralized Lightweight Group Key Management architecture for Access Control, presented in [1], attempts to solve the problems mentioned in the above analysis.

The article presents a secure data exchange protocol, the most critical component of the cryptographic key generation, renewal, and distribution (KGR) system. The developed security mechanisms use a hardware-based Trusted Platform Module (TPM) to establish local trust structures and ensure secure storage and data exchange on IoT network node resources. The developed protocol guarantees security regardless of the protection applied to the MQTT service used, which is an essential advantage for applications in federated environments.

II. CONCEPT OF A DATA EXCHANGE PROTOCOL

A. Characteristics of the cryptographic key generation and renewal system

The system of generation, renewal, and distribution of cryptographic keys (KGR) for a hybrid environment, in which a protected exchange of data between domains of sensor nodes and traditional systems (IT systems) will be required, is presented in figure Fig. 1.

Clients of the KGR system can be sensor node networks and traditional IT systems. Sensor node networks are organized as secure sensor node domains [9][10]. Only representatives (Gateway nodes) of secure sensor node domains and representatives of other systems participate in distributing cryptographic keys in the KGR system. Each gateway node works as a sink node for data originating from domain nodes or the IT system and as an emitter of data

□ This work was supported by the UGB-798 university project.

¹ <https://www.zymbit.com/zymkey>

originating outside for domain nodes or nodes of the IT system.

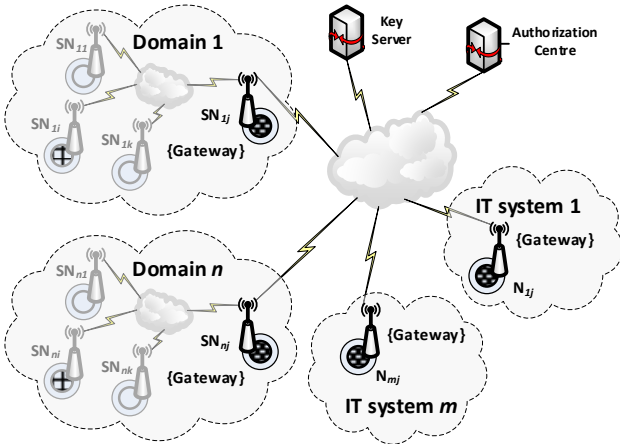


Fig. 1 The way various domains and IT systems cooperate with the key server

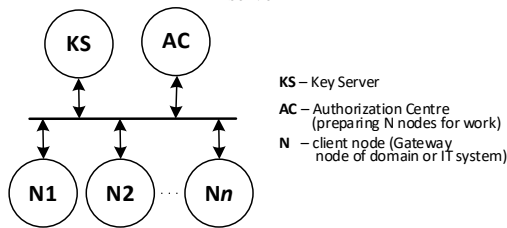


Fig. 2 Components of the KGR system

In the KGR system (Fig. 2), the source of cryptographic keys will be a separate node KS (Key Server) to which the N_1, N_2, \dots, N_n nodes (gateway nodes of domains and IT systems) will have access. The AC (Authorization Center) node is also a system component. It will be responsible for managing the KS node and adding new identifiers for the authorized N nodes in the KS node resources.

The MQTT protocol will be used to exchange data to facilitate key distribution for IoT networks. The protocol is event-driven, and data exchange between nodes is based on the publish /subscribe (Pub/Sub) pattern. The sender (Publisher) and receiver (Subscriber) are isolated from each other and use so-called "topics" to communicate. The intermediary of data exchange is a node called an MQTT broker.

B. Assumptions for the KGR system

Since one of the data exchange parties may be a mobile IoT network node, which is classified as a restricted device [11], it is assumed that the KGR system should satisfy the following assumptions:

- for the generation of symmetric cryptographic keys, the KGR system will use a high entropy random number generator (e.g., quantum random number generator),
- the KGR system is available on the Internet,
- the KGR system only handles requests from authorized clients,
- the KS node will obtain data about authorized clients from the AC node,

- the client can be implemented in hardware, but can also be a software component that is installed on a node that acts as a Gateway,
- the MQTT protocol will be used to distribute cryptographic keys,
- each KGR node uses a local trust structure that is constructed using the TPM module,
- sensitive data stored in the resources of each node and data transmission between nodes are cryptographically protected,
- each of the N-type nodes must be appropriately prepared and then registered in the KGR system before it can start normal working - registered nodes before they begin their activities are authenticated,
- a hardware TPM module supports all cryptographic procedures in the KGR system. TPM is an implementation of a standard developed by the Trusted Computing Group [12].

C. Procedures implemented in the KGR system

When creating a secure system and hardware and software configurations, secure procedures for using such a system and procedures for decommissioning it must be prepared. Procedures in the KGR system are implemented in two phases.

In the first phase, procedures are executed that prepare the KS node and N nodes for cooperation. These procedures include, but are not limited to, initializing the KS node, preparing credentials for the N nodes, and initializing and registering N nodes in the KS node resources.

In the second phase, procedures for generating and distributing symmetric keys for N nodes, renewing these keys, and secure data exchange between N nodes are performed. Only the procedures of the second phase will be described in detail later.

Fig 3 shows how data is exchanged between the nodes of the KGR system during the procedures mentioned above. The MQTT service is used in the KGR system only as an intermediary for data exchange and is neither the source nor the destination of the generated keys. Fig 4 shows the communication structure of the KGR system.

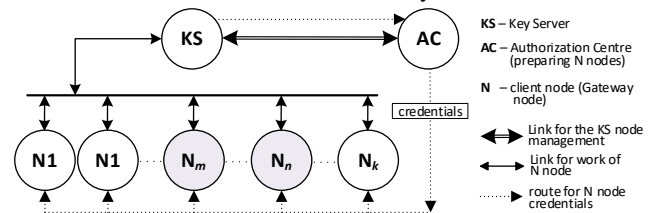


Fig. 3 The method of data exchange in the KGR system

MQTT service to exchange data requires defining a string called "topic". The main task of the MQTT broker is to forward messages published in a given "topic" by one client to clients subscribing to data with this "topic". To increase the security level in the KGR system, the content of each "topic" is random and known only to both parties of the

message exchange. In the description, strings related to the "topic" will be marked as TOPICn. Table 1 shows the purpose of the "topic" strings used.

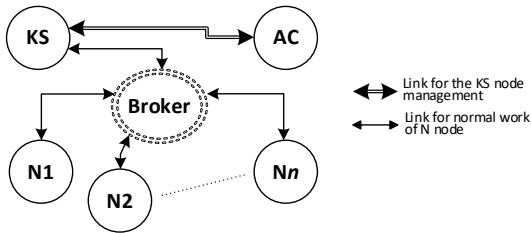


Fig. 4 The communication structure of the KGR system

TABLE 1.
LIST OF TOPICS USED BY THE NODES.

topic	node	purpose
TOPIC0	KS	for each N node for the first request during the registration procedure
TOPIC1	KS	for subsequent requests from the given N node during the registration procedure
TOPIC2	N	for requests from KS node (i.e., published by KS node)
TOPIC3 _m	N _m	for requests from N _m node (published by N _m node)
TOPIC4 _m	N _m	for publishing to N _m node

The rest of the paper describes the symmetric key generation and distribution procedure in detail.

III. DATA EXCHANGE PROTOCOL FOR THE KEY GENERATION AND DISTRIBUTION PROCEDURE

A. Procedure description

For simplicity in the following description, we will assume that the key set will be generated for a pair of nodes N1 and N2. Node N1 will initiate the key generation procedure.

The procedure for generating and distributing symmetric keys consists of three stages:

- a) requesting symmetric keys,
- b) delivery of symmetric keys to the requesting and destination nodes,
- c) confirmation of delivery of keys to the destination node.

This procedure requires the following conditions, which are provided in the first phase of KGR system preparation (not described here):

- the KS node is initialized and subscribes to TOPIC0 and TOPIC1 topics,
- the N-type nodes for which keys are to be generated must be registered in the KS node's resources, and each node subscribes to a TOPIC2 topic.

Meeting these requirements means that each N-type node has a local trust structure generated and has keys to secure data exchange between these nodes and the KS node.

An Encrypt-then-MAC (EtM) [13] approach requiring two keys will be used to secure data exchange. The AES algorithm will be used for encryption, and the HMAC algorithm for hash determination - both of which are supported on hardware by the TPM module. During one key generation operation, the node KS will prepare the

symmetric key NNSK (*Node to Node Security Key*), the initialization vector for this key, and the key NNSKsign (*Node to Node Security Key for signing*). Both these keys will be used only by nodes N1 and N2. A single key generation and distribution operation require performing the following actions:

- node N1 sends a request to node KS to generate a symmetric key pair for nodes N1 and N2,
- the KS node generates the NNSK key and the NNSKsign key and temporarily stores them,
- KS node sends the generated data to N1 and N2 nodes,
- the N2 node sends an acknowledgement of receipt of the keys to the N1 node via the KS node,
- after sending this acknowledgement, the KS node removes the NNSK and NNSKsign keys from its resources.

Fig. 5 shows the sequence diagram for these activities.

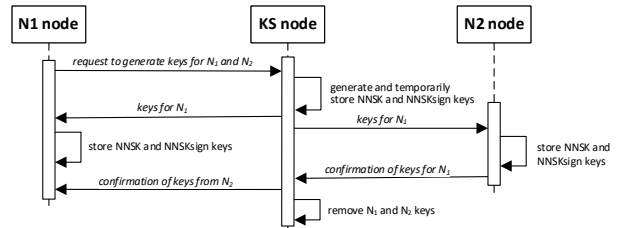


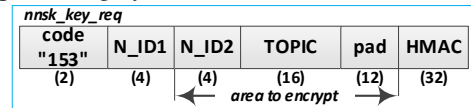
Fig. 5 Sequence diagram of key generation and distribution for node pair N1 and N2

The procedure results are the secure distribution of the generated keys and the contents of TOPIC3 and TOPIC4 to the N1 and N2 nodes. Nodes N1 and N2 will use the topics for subsequent secure data exchange among themselves using the MQTT service.

B. Frame structure description

The key generation and distribution procedure uses a data exchange protocol that utilizes six types of frames. For authenticated encryption data exchange will be used Encrypt-then-MAC (EtM) approach using the key pair, which are known only to the pair of nodes N_i and KS and was established during the registration procedure of node N_i. The list of these frames is as follows:

- **nnsk_key_req** - request to generate a new set of keys for N1 and N2 nodes - sending TOPIC generated by N1 for publishing by N2 node, which will act as TOPIC3,



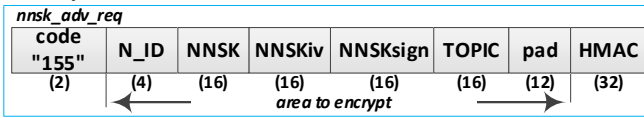
N_ID1 - ID of requesting N node
 N_ID2 - ID of destined N node
 TOPIC - Topics subscribed by N_ID1 node (TOPIC3)
 HMAC - Digital signature of the frame

- **nnsk_key_ans** - response to keys request - sending a set of keys and initialization vector,



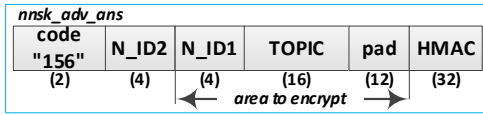
N_ID1 - ID of requesting N node
 N_ID2 - ID of destined N node
 NNSK - Node to Node Security Key + Initialization Vector
 NNSKsign - Node to Node Security Key for signing
 HMAC - Digital signature of the frame

- **nnsk_adv_req** - advertisement of generating a new set of keys on the initiative of node N1 - sending a set of keys, initialization vector, and TOPIC3,



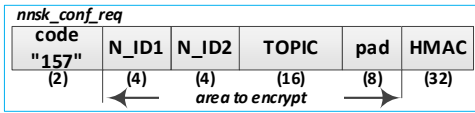
N_ID – ID of requesting N node (N_ID1)
 NNSK – Node to Node Security Key + Initialization Vector
 NNSKsign – Node to Node Security Key for signing
 TOPIC – Topics subscribed by requesting node (TOPIC3 by N_ID1)
 HMAC – Digital signature of the frame

- **nnsk_adv_ans** - confirmation of receipt of the set of keys - sending TOPIC generated by N2 for publishing by N1 node, which will act as TOPIC4,



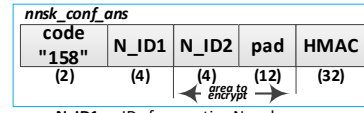
N_ID1 – ID of Secnd element from pair (N1, N2)
 N_ID2 – ID of requesting node (N1)
 TOPIC – Topics subscribed N2 node and for publishing by N1 node
 HMAC – Digital signature of the frame

- **nnsk_conf_req** - confirmation of delivery of a set of keys for N2 - sending TOPIC generated by N2 for publishing by N1 node,



N_ID1 – ID of requesting N node
 N_ID2 – ID of destined N node
 TOPIC – Topic subscribed by N2 node for publishing by N1 node
 HMAC – Digital signature of the frame

- **nnsk_conf_ans** - confirmation of the end of key distribution for a pair of nodes (N1, N2),



N_ID1 – ID of requesting N node
 N_ID2 – ID of destined N node
 HMAC – Digital signature of the frame

Fig. 6 shows the basic sequence diagram illustrating the operation of the data exchange protocol during the generation and distribution of the key set for a pair of nodes (N1, N2) for a use case in which the distribution procedure proceeds correctly. Diagram in Fig. 6 also includes the actions performed by the nodes involved in the data exchange during this procedure.

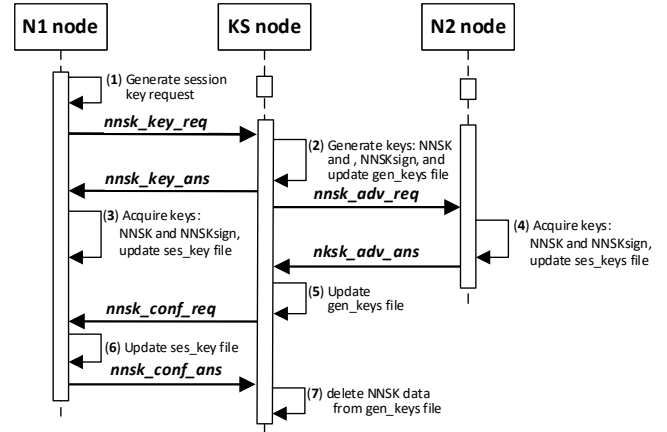


Fig. 6 Sequence diagram for key generation and distribution protocol

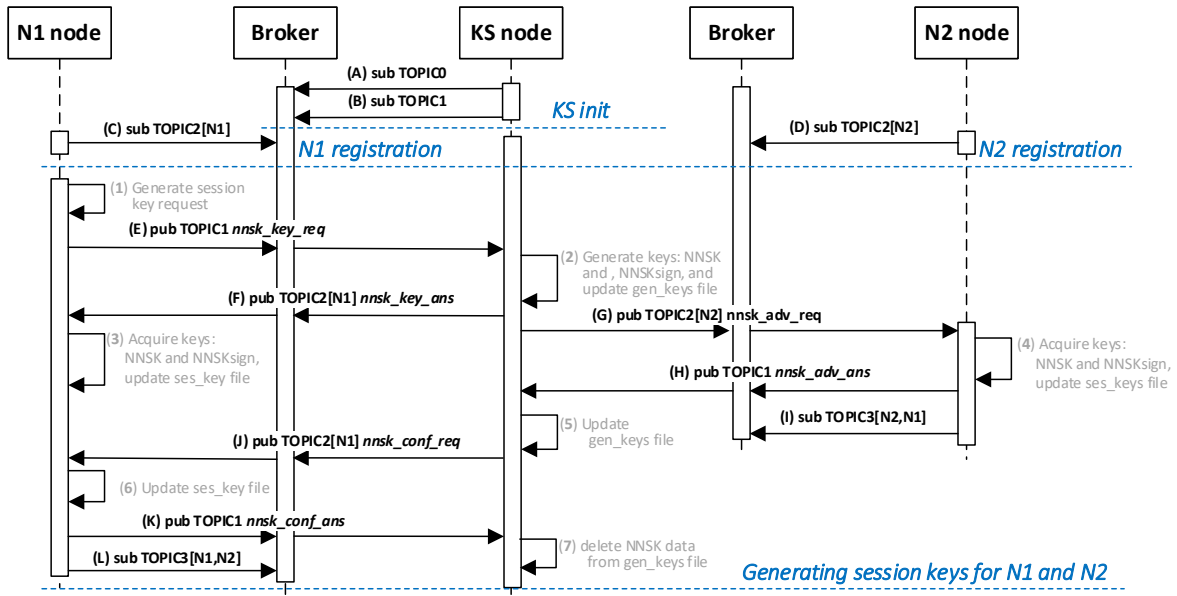


Fig. 7 The sequence diagram of the data exchange in the MQTT service for the generation and distribution key procedure.
 Legend: **sub TOPIC** - subscription to the topic broker "TOPIC", which is known to all nodes
sub TOPIC[Ni] - subscription to the "TOPIC" topic broker, which is known only to Ni and KS nodes
sub TOPIC[N1, N2] - subscription to the "TOPIC" topic broker, which is known only for N1 and N2 nodes
pub TOPICi nnsk_xxx_yyy - publishing a frame "nnsk_xxx_yyy" with the topic "TOPICi", where xxx={key, adv, conf}, yyy={req, ans}

In the MQTT service used to exchange data via the described protocol, a service broker plays the role of an intermediary. An important role is played in this service by appropriately using topics when sending data. Fig. 8 shows a sequence diagram illustrating data exchange in the MQTT service for the cryptographic key set distribution protocol.

C. Experiment description

To verify the operation of the KGR system, in particular the protocol for generating and distributing cryptographic keys, a lab bench was used that included four nodes (Fig. 9). One node played the role of KS, and two nodes played the role of N. The fourth node was a broker for the MQTT service. The experiments used Mosquitto 1.5.1 software, which implements the MQTT 3.1.1 protocol.

Each node of the lab bench was implemented in Python on a Raspberry Pi 3 platform with the Raspbian Buster system. The KS, N1, and N2 nodes were equipped with a Trusted Platform Module (TPM) 2.0 (the LetsTrust TPM). In the KGR system, TPM modules were the source of random numbers. It supported cryptographic procedures to generate keys, secure data exchange between nodes, and protect data stored in the resources of the KS, N1, and N2 nodes.

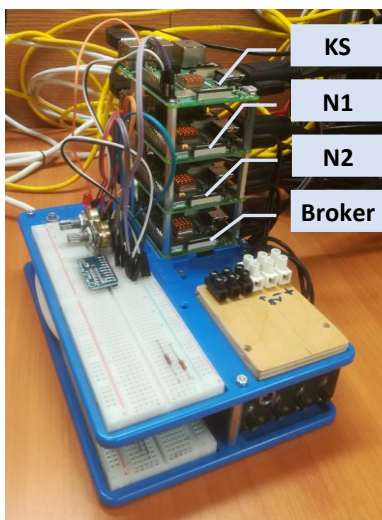


Fig. 8 View of lab bench for testing the KGR system.

During the testing of the KGR system, several experiments were conducted to confirm the correctness of the system's operation. They covered the procedures for preparing the KS node, distributing credentials for N-type nodes, registering N-type nodes in the KS node resources, and the procedure for generating and distributing keys for N-type nodes described in this paper.

IV. CONCLUSIONS

The described protocol for secure data exchange is the main component of the KGR system. This protocol provides secure distribution of cryptographic keys over the MQTT service IoT devices often use. This protocol can be used for

IoT devices and other IT systems that interact with IoT devices. Advantages of the protocol include:

- data exchange between the components of the KGR system is cryptographically secured,
- topics used in the MQTT service have random values and are known only to the nodes that use these topics - the KGR system also provides secure distribution of these topics,
- sensitive data of the KGR system nodes are cryptographically secured,
- the TPM module supports all cryptographic procedures,
- MQTT service broker is only an intermediary of data exchange and does not participate in any other way in procedures performed in the KGR system.

ACKNOWLEDGMENT

The presented study is the result of the author R&D activity in the IST-176 Research Task Group on Federated Interoperability of Military C2 and IoT Systems.

REFERENCES

- [1] Dammak, M.; Senouci, S.M.; Messous, M.A.; Elhdhili, M.H.; Gransart, C. Decentralized Lightweight Group Key Management for Dynamic Access Control in IoT Environments. *IEEE Trans. Netw. Serv. Manag.* 2020, 1–15, DOI: 10.1109/TNSM.2020.3002957.
- [2] Tan, H.; Chung, I. A Secure and Efficient Group Key Management Protocol with Cooperative Sensor Association in WBANs. *Sensors* 2018, 18, 3930, DOI: 10.3390/s18113930.
- [3] Zhong, H.; Luo, W.; Cui, J. Multiple multicast group key management for the Internet of People. *Concurr. Comput. Pract. Exp.* 2016, 29, e3817, DOI: 10.1002/cpe.3817.
- [4] Ding, W.; Hu, R.; Yan, Z.; Qian, X.; Deng, R.H.; Yang, L.T.; Dong, M. An Extended Framework of Privacy-Preserving Computation with Flexible Access Control. *IEEE Trans. Netw. Serv. Manag.* 2020, 17, 918–930, DOI: 10.1109/TNSM.2019.2952462.
- [5] Mehdizadeh, A.; Hashim, F.; Othman, M. Lightweight decentralized multicast-unicast key management method in wireless IPv6 networks. *J. Netw. Comput. Appl.* 2014, 42, DOI: 10.1016/j.jnca.2014.03.013.
- [6] Kung, Y.; Hsiao, H. GroupIt: Lightweight Group Key Management for Dynamic IoT Environments. *IEEE Internet Things J.* 2018, 5, 5155–5165, DOI: 10.1109/JIOT.2018.2840321.
- [7] Abdmeziem, M.R.; Tandjaoui, D.; Romdhani, I. A Decentralized Batch-Based Group Key Management Protocol for Mobile Internet of Things (DBGK). In *Proceedings of the IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*, Liverpool, UK, 2015; 1109–1117, DOI: 10.1109/CIT/IUCC/DASC/PICOM.2015.166
- [8] Yao, W.; Han, S.; Li, X. LKH++ Based Group Key Management Scheme for Wireless Sensor Network. *Wirel. Pers. Commun.* 2015, 83, 3057–3073.
- [9] J. Furtak, Z. Zieliński and J. Chudzikiewicz, Procedures for sensor nodes operation in the secured domain, *Concurr. Comput. Pract. Exp.* 2019, 32, e5183, DOI: 10.1002/cpe.5183.
- [10] J. Furtak, Z. Zieliński and J. Chudzikiewicz, A Framework for Constructing a Secure Domain of Sensor Nodes, *Sensors* 2019, 19, 2797, DOI: 10.3390/s19122797.
- [11] Borman C., Ersue M., Keranen A., Terminology for Constrained-Node Networks, RFC 7228, Internet Engineering Task Force (IETF), May 2014.
- [12] Trusted Computing Group. TPM Main Part 1 Design Principles. Specification Version 1.2, Revision 116; Trusted Computing Group: Beaverton, OR, USA, 2011.
- [13] Information technology - Authenticated encryption. 1977:2020. ISO/IEC. Retrieved November 2020.

Formal verification of security properties of the Lightweight Authentication and Key Exchange Protocol for Federated IoT devices

Michał Jarosz

Cybernetics Faculty

Military University of Technology

Warsaw, Poland

michal.jarosz@wat.edu.pl

Konrad Wrona

NATO Cyber Security Centre /

Military University of Technology

The Hague, Netherlands / Warsaw, Poland

konrad.wrona@[ncia.nato.int,wat.edu.pl]

Zbigniew Zieliński

Cybernetics Faculty

Military University of Technology

Warsaw, Poland

zbigniew.zielinski@wat.edu.pl

Abstract—The federated nature of many crucial Internet of Things (IoT) applications introduces several challenges from a security perspective. To address critical challenges related to the authentication and secure communication of IoT devices operating in federated environments, we propose a new authentication and key exchange protocol based on a distributed ledger. Our protocol uses the unique configuration fingerprint of an IoT device and does not require secure storage in participating IoT devices. To validate the correctness of our design, we have performed formal modeling and verification of the security properties, using two different verification tools: Verifpal and the Tamarin prover.

I. INTRODUCTION

THE APPLICATIONS of Internet of Things (IoT) expand rapidly to many mission-critical areas, such as smart health care and Humanitarian Assistance and Disaster Relief (HADR) [1]. Many of these applications are based on the concept of *federation* in which IoT devices operated by different federated partners must communicate with each other securely.

Many protocols are used for authentication and key exchange between devices described in the literature. An overview of such protocols is presented in [2]. The proposed solutions are based on a variety of approaches, including Public Key Infrastructure (PKI) [3], blockchain [4, 5, 6], and shared keys [7, 8]. An advantage of our proposal is its focus on the federated environment, where devices belonging to one organization (and regulated by its security policy) can be accessed and used by other organizations belonging to the federation.

Federating separate IoT administrative domains introduces several challenges from a security perspective. One of the fundamental challenges is establishing an effective identity and access management (IAM) framework, which is necessary to ensure trust and secure communication between federated IoT devices [9]. The particularly critical IAM challenge is the authentication and authorization of federated IoT devices.

To address these challenges, we propose in Section II a Lightweight Authentication and Key Exchange Protocol for Federated IoT (LAKEPFI) that supports flexible authenticated

key exchange based on Hyperledger Fabric (HLF) [10]. The solution uses the unique configuration fingerprint of an IoT device and does not require secure storage space in participating devices. Our protocol enables secure communication of heterogeneous federated IoT devices, including devices equipped with additional security hardware, such as Physical Unclonable Functions (PUF) [11], and devices characterized by very limited computing resources such as Arduino.

Constructing a new security protocol is an error-prone process. Therefore, it is essential to verify its security properties using various techniques. The process of verifying the security properties of protocols is a common operation. Any protocol used to communicate with devices must undergo this process. In this article, we will focus on describing the verification of the security properties of the developed protocol. For this purpose, we used two tools: Verifpal and Tamarin. These tools have previously been used for the verification of some commonly used IoT communication protocols [12].

In Section III we briefly introduce formal modeling and verification approaches that can be used to validate the security properties of LAKEPFI. In Sections IV and V, we present and discuss formal modeling and verification of the security properties of LAKEPFI using Verifpal and the Tamarin prover. In particular, our aim is to prove formally that the designed protocol provides: 1) message secrecy; 2) message authentication; 3) freshness.

II. LIGHTWEIGHT AUTHENTICATION AND KEY EXCHANGE FOR FEDERATED IOT DEVICES

A. Federated IoT architecture

The main goal of the proposed protocol is to establish efficient and secure communication between devices belonging to different organizations. The main participants in this protocol are IoT devices that belong to other organizations and services on the organization's side: the application gateway and distributed ledger.

Fig. 1 shows the general architecture of the components that are used in the LAKEPFI.

The IoT device consists of two components:

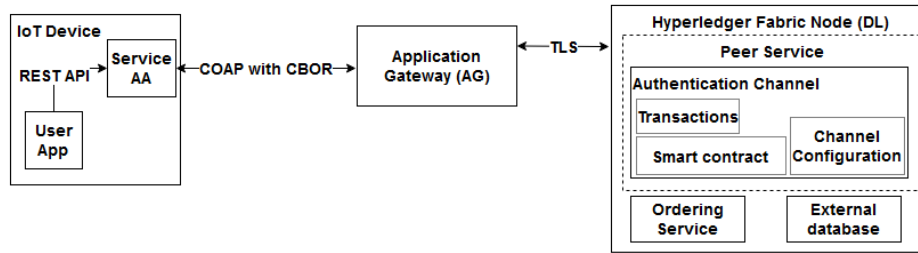


Fig. 1. Component architecture and interconnection scheme.

- 1) AA Service (Authentication and Authorization Service), responsible for communicating with the application gateway using the solution we describe;
- 2) User App, the user application that connects to the AA service is agnostic to our framework. Each user application uses a unified interface provided by the AA service.

Our protocol is an application layer protocol, and although it can be used in combination with existing lower-level security protocols to provide increased defense-in-depth, it is agnostic to these lower-layer protocols. In our proof of concept, for efficiency reasons, Constrained Application Protocol (CoAP) [13] with Concise Binary Object Representation (CBOR) [14] is used for communication between the device and the application gateway (AG). Communication between the application gateway and distributed ledger nodes uses Transport Layer Security (TLS). The application gateway is responsible for the following:

- 1) Conversion from CoAP to TLS and vice versa, due to the fact that Hyperledger Fabric requires TLS with certificates, while IoT devices must use lightweight protocols;
- 2) Filtering out of invalid and malicious messages;
- 3) Load balancing, therefore, ensuring that the IoT device does not need to know the whole distributed ledger architecture, which can change.

The last component is the distributed ledger node, which stores all smart contracts and has access to all channels. In our implementation, we used Hyperledger Fabric as a distributed ledger implementation. The Hyperledger Fabric Node consists of two services:

- 1) *Peer*, responsible for the verification and recording of transactions in the distributed ledger;
- 2) *Orderer*, responsible for the creation of a block.

B. Description of the protocol

The proposed protocol is based on the use of unique parameters of the IoT device for its authentication. Within this protocol, three main methods are described:

- 1) Registration of an IoT device in a distributed ledger;
- 2) Communication of an IoT device with a distributed ledger node;

- 3) Communication between IoT devices, especially those belonging to different organizations within the federation.

C. Registration phase

The first operation is device registration. This is required for the device to communicate with other devices and services. The registration process can be performed in two different ways: 1) connect directly to the application gateway; or 2) put the device at the destination and secure the communication for the duration of the registration with a one-time login and password. During the registration phase, the device sends values to the application gateway for the parameters that define the device. Each parameter should have an appropriate entropy. This is a kind of fingerprint from the IoT device. Depending on the capabilities of the IoT device, the parameters recorded in the array may include the following:

- 1) Unique configuration data - in this case, the unique data of the device is used. It can be hardware data, e.g., device serial number, memory card serial number, etc., and it can be software data, e.g., partition IDs, file system IDs, keys stored on the device,
- 2) PUF data - data from the PUFs embedded in the IoT device,
- 3) Random strings (for example, keys) - in this case, the device generates random strings with required entropy and needs to store them securely. This method is used only if neither 1) nor 2) can be used.

In the first and second cases, the device does not store the key in its storage; it is enough to hold a program to obtain values of specific parameters. The device performs an appropriate operation to obtain parameters, e.g., a system command to obtain a particular parameter. The second and third cases are prepared for devices with a minimum of 50 kB RAM and 250 kB flash memory, that is, the so-called class C2 [15]. However, in case 2, the device must support PUF.

In the registration process, a set of parameters P_A consisting of its parameters $\{p_1, p_2, \dots, p_n\}$ is written in the distributed ledger DL . The communication diagram is shown in Fig. 2. As written in the beginning of this section, there are two options for securely sending P_A from the IoT device A to the application gateway AG . After receiving the array P_A , the application gateway AG generates a new identifier ID_A for the device A . The ID_A along with the parameter array P_A is stored in the

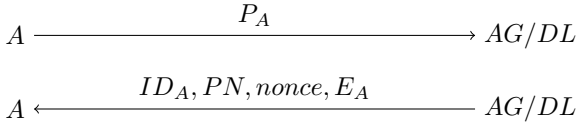


Fig. 2. Communication between the IoT device and a ledger node during the registration phase

distributed ledger DL . After the success of storing these data in the ledger DL , a response is generated to the IoT device A . In the response, the identifier ID_A and the current timestamp t are sent to the device ID_A in encrypted form E_A . The encryption key K is generated by the distributed ledger node DL_i . This key K is the result of a hash function from the concatenation of a subset $\{1, \dots, n\}$: $PN_{DL \rightarrow A} = \{n_1, n_2, \dots, n_k\}$ of the parameters $K = H(p_{n_1} | p_{n_2} | \dots | p_{n_k})$ sent by the device. The parameters used for encryption are chosen at random. In addition to ciphertext $E_A = E[K; nonce; (ID_A, t)]$, where $E[\cdot]$ is the encryption function to encrypt (ID_A, t) using K and $nonce$, AG also sends the identifier in plaintext ID_A , the nonce value $nonce$, and the identifiers of the parameters $PN = \{p_{n_1}, p_{n_2}, \dots, p_{n_k}\}$, whose values were used to create the key K , are also sent.

Based on the number of parameters $PN = \{p_{n_1}, p_{n_2}, \dots, p_{n_k}\}$ that are sent in the response, the device knows which parameters were used to generate the key K so it can recreate the key K and thus decrypt the message. After decryption, it checks if the timestamp t sent in reply differs more than 5 seconds from the current one. Then it checks if the identifier ID_A sent in plaintext and the ciphertext E are the same. If all these operations were successful, the device is registered and then uses the received identifier ID_A and the generated set of parameters P_A .

D. Communication procedure between IoT device and distributed ledger node

When a device A communicates with an application gateway AG that will perform a task, the device A encrypts the data $data$ it wants to send to the gateway AG in the same way as described for the process of generating a response from the ledger DL during device registration. The communication diagram is shown in Fig. 3. The device A selects a subset of parameters $\{1, \dots, n\}$: $PN_{A \rightarrow DL} = \{n_1, n_2, \dots, n_k\}$ and generates a key $K_{A \rightarrow DL} = H(H(p_{n_1}) | H(p_{n_2}) | \dots | H(p_{n_k}))$ from them. The key $K_{A \rightarrow DL}$ together with the random generated nonce value $nonce_{A \rightarrow DL}$ is used to encrypt the data $data$ and the current timestamp $t_{A \rightarrow DL}$: $E_{A \rightarrow DL} = E[K_{A \rightarrow DL}; nonce_{A \rightarrow DL}; (data, t_{A \rightarrow DL})]$. To the application gateway AG , device AA sends: an identifier ID_A , a nonce $nonce_{A \rightarrow DL}$, parameter numbers $PN_{A \rightarrow DL}$ that define the parameters used to generate the key $K_{A \rightarrow DL}$ and a ciphertext $E_{A \rightarrow DL}$. The application gateway AG sends these data to the distributed ledger node DL_i to decrypt the ciphertext $E_{A \rightarrow DL}$. The distributed ledger node DL_i is capable of generating a key $K_{A \rightarrow DL}$ based on the number of parameters $PN_{A \rightarrow DL}$ and

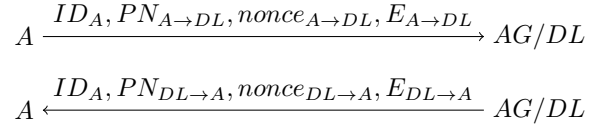


Fig. 3. Communication between an IoT device and the distributed ledger

the identifier ID_A . Using this key $K_{A \rightarrow DL}$ and the nonce value $nonce_{A \rightarrow DL}$ it performs decryption of the ciphertext $E_{A \rightarrow DL}$. The timestamp $t_{A \rightarrow DL}$ stored in the ciphertext $E_{A \rightarrow DL}$ and the current one are verified. If the ciphertext $E_{A \rightarrow DL}$ was successfully decrypted and the timestamp $t_{A \rightarrow DL}$ is correct, the application gateway AG receives the encrypted $data$. Based on $data$, it determines what should be executed and performs the requested operation. The response $response$ is generated in the same way. The device A receives the identifier ID_A , the new nonce value $nonce_{DL \rightarrow A}$ used in the response $response$ encryption process, the parameter numbers $PN_{DL \rightarrow A}$ that define the parameters used to generate the key $K_{DL \rightarrow A}$ and the ciphertext $E_{DL \rightarrow A} = E[K_{DL \rightarrow A}; nonce_{DL \rightarrow A}; (response, t_{DL \rightarrow A})]$. The device A after receiving the message is capable of generating the key $K_{DL \rightarrow A}$ and thus decrypting the ciphertext $E_{DL \rightarrow A}$ and getting the response $response$.

E. Communication procedure between IoT devices

To communicate with each other, it is necessary to authenticate the devices that want to communicate with each other and verify that such communication is authorized. The communication diagram is shown in Fig. 4. The device that wants to establish communication is the device A with ID_A . This device sends the ID_B of the device B to the application gateway AG . The device A can get the ID_B of device B in many ways, e.g., it can receive the ID_B of device B from another device, it can have a record that needs to communicate with that device, or device B can announce that it has a certain type of information. The identifier ID_B is secured in a way that is identical to the data in the device communication process with the distributed ledger DL described in the previous subsection. The application gateway AG sends the request to decrypt to the distributed ledger node DL . The node DL returns the decrypted identifier ID_B . The application gateway AG then sends a request to create a key $K_{A \leftrightarrow B}$ for communication between devices A and B . DL verifies based on the rules stored in the DL authorization channel whether A and B can communicate. If so, then it generates the key $K_{A \leftrightarrow B}$. To create a key $K_{A \leftrightarrow B}$, DL selects a subset of k -elements of $p_{n_1}, p_{n_2}, \dots, p_{n_k}$ from the parameter array P_B of the device B , which is stored in the ledger. The key $K_{A \leftrightarrow B}$ is constructed similarly to the previous processes; only in this case is the current timestamp $t_{A \leftrightarrow B}$ is also included: $K_{A \leftrightarrow B} = H(p_{n_1} | p_{n_2} | \dots | p_{n_k} | t_{A \leftrightarrow B})$. The response to device A must send the key $K_{A \leftrightarrow B}$, timestamp $t_{A \leftrightarrow B}$, and parameter numbers $PN_{A \leftrightarrow B}$ that were created to create the key $K_{A \leftrightarrow B}$. The response is secured in the same way as in the previous

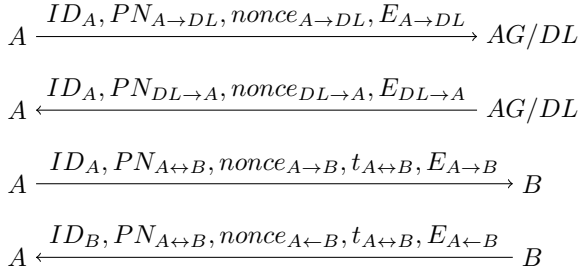


Fig. 4. The procedure of communication between IoT devices

cases. The device A decrypts the message in the same way as in the previous case. If everything is correct, the device A can now send *data* to the device B . To do this, it has to generate a new nonce $nonce_{A \rightarrow B}$. Using the key $K_{A \leftrightarrow B}$ received in the response and the nonce value $nonce_{A \rightarrow B}$, it can encrypt *data* and the timestamp $t_{A \leftrightarrow B}$ that will be sent to the device B : $E_{A \rightarrow B} = E[K_{A \leftrightarrow B}; nonce_{A \rightarrow B}; (data, t_{A \leftrightarrow B})]$. Then device A sends to device B : $ID_A, PN_{A \leftrightarrow B}, t_{A \leftrightarrow B}, nonce_{A \rightarrow B}, E_{A \rightarrow B}$.

The device B after receiving the message can decrypt the ciphertext $E_{A \rightarrow B}$ because the key $K_{A \leftrightarrow B}$ was created by the distributed ledger node DL based on the parameters P_B of the device B and the timestamp $t_{A \leftrightarrow B}$. Using the key $K_{A \leftrightarrow B}$ and the nonce $nonce_{A \rightarrow B}$, the device B decrypts the ciphertext $E_{A \rightarrow B}$. If device B successfully decrypts the ciphertext $E_{A \rightarrow B}$, this means that the key $K_{A \leftrightarrow B}$ has been created by the distributed ledger node DL and, therefore, has been issued to the authorized entity A . To secure the response, the device B uses the same key $K_{A \leftrightarrow B}$ but with a new nonce value $nonce_{A \leftarrow B}$. Furthermore, the new timestamp $t_{A \leftarrow B}$ is generated in ciphertext. The new ciphertext is built: $E_{A \leftarrow B} = E[K_{A \leftrightarrow B}; nonce_{A \leftarrow B}; (result, t_{A \leftarrow B})]$. Therefore, the answer contains: $ID_B, PN_{A \leftrightarrow B}, nonce_{A \leftarrow B}, t_{A \leftrightarrow B}, E_{A \leftarrow B}$.

III. MODELING AND VERIFICATION

Building a secure communication protocol requires verification of its security properties. Such properties include, for example, authentication of the communicating parties, the confidentiality and integrity of the transmitted data, and the equivalence property; that is, an attacker cannot distinguish between any two stages of the protocol.

Formal verification can confirm (or deny) that the protocol is secure. A protocol can be mathematically formalized by [16]:

1) *Symbolic (Dolev-Yao [17]) model*, in which cryptographic primitives are represented as black boxes, messages are terms in these primitives, and the attacker is restricted to using these primitives. This model makes the execution of automatic proofs relatively easy. The symbolic model is robust to attacks if no attack can occur for each trace. By trace, we mean one run of the protocol. Research progress on this type of model is considered highly advanced.

2) *Computational model*, in which messages are strings of bits, cryptographic primitives are functions that operate on these strings, and the attacker is any probabilistic Turing machine. This mode is generally used by cryptographers. A computational model is robust to attacks if no attack can occur for every trace except those with a negligibly small probability. The computational model is more realistic, but still, since it is only a model, it will not give us an answer to whether the developed protocol is resilient to some attacks, such as side-channel or physical attacks. In the case of computational models, most of the time, cryptographic properties have to be proven manually. The proofs in this model are more difficult to perform and analyze. This type of modeling is much less mature. Therefore, we did not use this type of modeling.

It is worth mentioning that, regardless of the modeling approach used, even minor changes to the protocol require a revision of the model used for the analysis. Examples of tools that use the symbolic model include:

- 1) AVISPA [18] uses a modular and expressive High-Level Protocol Specification Language (HLPSL). It supports many back-end tools to verify a model, and the result of the model verification is easy to interpret. The implemented techniques offer protocol analysis, such as protocol falsification (by finding an attack on the input protocol) and abstraction methods for finite and infinite sessions, among others. AVISPA is currently not widely used in papers.
- 2) Verifpal [19] uses an intuitive language to describe the model and the result is easy to interpret. Verifpal can validate forward secrecy, key compromise, impersonation, and other advanced queries. It also supports unbounded sessions, fresh and random values, and other valuable features of the symbolic model. However, users cannot define their primitives. It is one of the youngest tools and is still being actively developed.
- 3) ProVerif [20], like the previous tools, provided predefined and reusable primitives. However, contrary to AVISPA and Verifpal, new primitives can also be defined. The way of modeling in ProVerif is similar to the approach used in Verifpal and AVISPA, but ProVerif uses a different verification method. If ProVerif checks that a security property is fulfilled, then it is indeed so, but ProVerif cannot always prove the properties of the tested protocol.
- 4) Tamarin prover [21] in its capabilities and popularity is similar to ProVerif; Tamarin prover has a different way of modeling protocols than the other tools. In the case of the Tamarin prover, we model the rules of transition between states in the protocol, while every operation within the protocol is modeled in other protocols. Each rule in Tamarin prover has an input (describes what it takes in), an output (what the result is), and facts. The proof in the case of Tamarin is to verify whether the

lemma we describe is true or not. Similarly to ProVerif, the Tamarin prover is not always able to prove the properties of the tested protocol.

For the analysis of our protocol, we have selected Verifpal and Tamarin. These formal verification tools are described in more detail in the next section. Both Verifpal and Tamarin are widely accepted and are currently used by researchers and practitioners to analyze security protocols. Verifpal supports fast modeling of a protocol and verification of its basic properties (message secrecy and authentication). On the other hand, the Tamarin prover is a more powerful tool that can allow us to verify a protocol in more detail. Both tools also verify the model automatically. Note that these tools work with an unbounded number of sessions, making the problem undecidable [16].

When modeling a security protocol, several issues need to be taken into account [22]:

- 1) One should assume the correctness of low-level cryptography mechanisms; for example, when using encryption or random value generation, one should assume that these functions work correctly.
- 2) Appropriate assumptions should be made about external services, e.g., network services such as time servers and trusted third parties.
- 3) Since protocol messages often need to pass through various middle boxes, such as firewalls and guards, and the protocol participants may be mobile or connected via unreliable communication channels, it has to be assumed that the messages may have different forms or may not reach the other participant.
- 4) Some protocols require early negotiation to determine the cryptographic primitives to be used, which affect the level of security and performance.
- 5) The reaction of participants to the occurrence of an error should be appropriately modeled, as proper error handling affects the security of the protocol.

Verifpal offers a more native way of writing the protocol, making it more readable and easier to learn. Verifpal defines two types of an attacker. The first type is *active attacker*, which can intercept all protocol messages, modify them, and inject his messages. The second type is *passive attacker*, which can only intercept messages sent between protocol participants. An attacker is free to use any combination of actions available to him. Verifpal has defined cryptographic primitives such as (a)symmetric encryption, authenticated encryption [23], Diffie-Hellman key exchange, digital signature, and a secure hash function. Verifpal does not allow for the definition of additional cryptographic primitives. The verification result is easy to interpret; precise information about what has been changed and what property is not met. Verifpal is one of the few tools that examines the unlinkability property. Unlinkability is a situation where for two observed values, the adversary cannot distinguish between a protocol execution in which they belong to the same user and a protocol execution in which they belong to two different users [19].

Tamarin prover offers automatic and interactive theorem proving models [21]. Although the Tamarin prover automatically generates proofs, user interaction is sometimes required. The result of the proof itself also brings inconclusiveness to the problem. The analysis is based on labeled multiset rewriting rules to specify protocols and adversary capabilities, a guarded fragment of first-order logic to determine security properties and functions, and equational theories to model the algebraic properties of cryptographic protocols. Furthermore, all events related to security properties are annotated with a time point $t \in Q$, and a basic comparison of time points can be used. Tamarin prover applies a constraint-solving algorithm based on backward search and heuristics that attempts to validate or falsify security properties. This approach requires significant knowledge and experience in formal modeling [24]. Therefore, modeling the protocol using the Tamarin prover is complex, and the result of model verification is also rather difficult to interpret.

IV. VERIFPAL

Modeling in the Verifpal tool first involves specifying the participants (parties) who will use the protocol. In the model itself, we define what operations a participant will perform and what data are transmitted between them. Finally, there are questions about the security properties of the data transmitted between participants. The role of Verifpal is to confirm the secrecy of transmitted data, confirm its authentication, and verify its freshness. Using Verifpal, three LAKEPFI operations were modeled:

- 1) Device registration,
- 2) Communication IoT device with Hyperledger Fabric,
- 3) Communication IoT device with other IoT device.

In this article, we will only describe in detail the operation of communicating an IoT device with a Hyperledger Fabric node because the other operations are very similar to this one. However, we will discuss model verification results for all three operations at the end of this subsection. During modeling, we followed the rules described in the III section.

First, we configure the mode in which the attacker should operate. In our case, an active attacker can influence the messages sent. Then, we define what the device knows. The device *DeviceA* knows its *ID*, which is public, so it also knows *HLF* (Hyperledger Fabric node) and the attacker. Additionally, the device knows the secret data *dataA* it wants to send. In the real-life protocol flow, the device generates the parameter numbers *paramnumbersAS* that are used to create the key. However, in the case of the protocol model, these values are not generated, but are sent and therefore must be included in the message.

The device generates timestamp *timestamp* and nonce *nonceAS*. In a real-life protocol flow, the device would now generate the *keyAS* key based on the set of specific parameter values. However, Verifpal does not allow this operation. Therefore, *keyAS* is generated as a random value. The device concatenates *dataA* and *timestamp*. The result of this

concatenation is then encrypted using *keyAS* and *nonceAS*. The result of encryption is the ciphertext *EdataAS*.

```
attacker[active]
principal DeviceA[
knows public ID
knows private dataA
generates paramnumbersAS
generates timestamp
generates nonceAS
knows private keyAS
dataToEncryptAS = CONCAT(dataA , timestamp)
EdataAS =
  AEAD_ENC(keyAS , dataToEncryptAS , nonceAS)
]
```

The *DeviceA* sends to *HLF* the *paramnumbersAS*, *nonceAS* and the ciphertext *EdataAS*. This information is learned by the attacker at this point.

```
DeviceA -> HLF: paramnumbersAS , nonceAS , EdataAS
```

HLF also knows the key *keyAS*, which should normally be generated using *paramnumbersAS* and a set of parameters. Then using *keyAS* and *nonceAS* it decrypts *EdataAS*. The result is split, leading to the data *DataAS* and the timestamp *timestampFromA*. In real implementation, the current timestamp is compared with *timestampFromA*, while in Verifpal, the ASSERT function does not affect anything. *HLF* generates a response *reply*, which will be sent to *DeviceA*. It also generates a new timestamp *timestampToA* and *paramnumbersSA* that, like *paramnumbersAS*, are not used but are sent. The *reply* is concatenated with *timestampToA*. The result is encrypted using a new key *keySA* and a new value *nonceSA*. A ciphertext *EdataSA* is created. As before, the new key is generated as if it were a random value, rather than derived from *paramnumbersSA* and the parameter array. Of course, the key *keySA* is marked as private, so it is not known to the attacker.

```
principal HLF[
knows private keyAS
DdataAS = AEAD_DEC(keyAS , EdataAS , nonceAS)
DataAS , timestampFromA = Split(DdataAS)
generates timestampNow
_ = ASSERT(timestampFromA , timestampNow)
generates reply
generates timestampToA
generates paramnumbersSA
dataToEncryptSA = CONCAT(reply , timestampToA)
generates nonceSA
knows private keySA
EdataSA =
  AEAD_ENC(keySA , dataToEncryptSA , nonceSA)
]
```

HLF sends *paramnumbersSA*, *nonceSA*, and *EdataSA* to *DeviceA*. At this point, the attacker learns these values.

```
HLF -> DeviceA: paramnumbersSA , nonceSA , EdataSA
```

The final step is for *DeviceA* to decrypt the response. For this, *DeviceA* knows the key *keySA*, which, as in the previous steps, is not created from *paramnumbersSA* and the parameter array. The ciphertext *EdataSA* is decrypted using *keySA* and *nonceSA*. Finally, the timestamp contained in the ciphertext is compared with the current one.

```
principal DeviceA[
knows private keySA
DdataSA = AEAD_DEC(keySA , EdataSA , nonceSA)
DataSA , timestampFromHLF = Split(DdataSA)
generates timestampNowReply
_ = ASSERT(timestampNowReply , timestampFromHLF)
]
```

The very end of the above listing defines queries concerning the properties that Verifpal has to check. For each operation, there are three properties: confidentiality of transmitted data, authentication of transmitted data, and freshness.

```
queries[
confidentiality? dataToEncryptSA
confidentiality? dataToEncryptAS
authentication? DeviceA -> HLF: EdataAS
authentication? HLF -> DeviceA: EdataSA
freshness? EdataAS
freshness? EdataSA
]
```

When verifying the registration operation and communication of the IoT device with the Hyperledger Fabric node, Verifpal found no errors and therefore confirmed confidentiality, authentication, and freshness.

On the other hand, for the third operation, communication between IoT devices, Verifpal found two errors: failure to authenticate *HLF* and lack of freshness when sending data from *HLF* to *DeviceA*. However, both errors are false positives. In both cases, Verifpal sets the value of *nonceSA* to nil (null), and this situation in a real run will return an error and *DeviceA* will reject the message. As we mentioned in Section III, modeling does not assume error handling, and here we have a good example.

In Verifpal, we had to apply some simplifications:

- 1) The key is a generated value, not a value generated from the parameters of device;
- 2) The parameters themselves are also a generated value and not a list with numbers;
- 3) There is no way to compare timestamps (although there is an ASSERT function, which is not used in Verifpal).

V. TAMARIN PROVER

The approach to modeling in Tamarin prover is quite different from that in Verifpal. For the Tamarin prover, think of a protocol as a set of states where information not passed to another state is forgotten. The conditions themselves are written as lemmas, which the Tamarin prover verifies. As with Verifpal, Tamarin prover allowed us to confirm secrecy, authenticity, and freshness. In the case of the Tamarin prover, three operations have been modeled:

- 1) Device registration,
- 2) Communication IoT device with Hyperledger Fabric,
- 3) Communication IoT device with other IoT device.

As in Section IV, we will describe in detail the operation of communicating an IoT device with a Hyperledger Fabric node. At the end of this section, we present the results for both operations.

First, we need to define the operations that we will use. The value after / indicates the number of arguments that the

operation will take. In addition, two functions are defined: decrypt and verify.

```
theory Device2Ledge
begin
functions:
aead/3, decrypt/2, verify/3, true/0
equations: decrypt(aead(k, p, a), a, k)=p
equations: verify(aead(k, p, a), a, k)=true
```

In the first rule, we generate a key that is shared between two parties. The function Fr means that key is random. Like Verifpal, here there is no possibility of generating the key as described in the actual implementation. In the set of facts, we specify that $Client$ and HLF know our key. We send the associated client $Client$ with the key key and the HLF node HLF with the same key to the next state.

```
rule Setup:
[ Fr(~key) ]
--[ID_Client($Client,~key),
  ID_Server($HLF,~key)]->
[!Identity($Client, ~key),
 !Identity($HLF, ~key)]
```

The next state takes in a new nonce value $nonce$ that is generated on input to this state, and two bindings, $Client$ and HLF with the key key . In the set of facts, we have written that there is communication between the client and the server using the key key and the value $nonce$. The output is a state in which we send a message from $Client$ to HLF with $nonce$ and the ciphertext $aead(key,'Data', nonce)$.

```
rule Client_1:
[ Fr(~nonce), !Identity($Client, key),
 !Identity($HLF, key) ]
--[Client2HLF($Client,$HLF,
 <key,~nonce>)]->
[ Out(<$Client,$HLF,~nonce,aead(key,
 'Dane',~nonce)>)]
```

The next rule defines a state that takes as input the newly generated nonce value $nonce2$, the binding of HLF to key , and the message from the previous state. In the set of facts, we define that there is communication between HLF and $Client$ using key and $nonce2$, and we verify the message. When leaving this state, we send a message from HLF to $Client$ with $nonce2$ and a response $reply$ encrypted with key and $nonce2$.

```
rule HLF_1:
let EncMessage = aead(key,'Dane',nonce)
in
[ Fr(~nonce2),!Identity($HLF, key),
 In(<$Client,$HLF,nonce,EncMessage>) ]
--[ HLF2Client($HLF,$Client,
 <key,~nonce2>),
  Eq(verify(EncMessage,nonce,key),true)
 ]->
[ Out(<$HLF,$Client,~nonce2,
 aead(key,'Respond',~nonce2)>)]
```

In the last rule, we define that the state receives as input a binding between $Client$ and key , as well as a message from HLF to $Client$. In the set of facts, we verify the message. This state is the final state.

```
rule Client_2:
```

```
let EncMessageFromHLF =
  aead(key,'Respond',nonce2)
in
[ Fr(~nonce3),!Identity($Client, key),
 In(<$HLF,$Client,nonce2,EncMessageFromHLF>) ]
--[Eq(verify(EncMessageFromHLF,nonce2,
 key),true)]->
[ ]
```

In addition, we have defined some restrictions. The first two restrictions specify that $Client$ cannot communicate with $Client$ and HLF cannot communicate with HLF . In $DenyClient2Client$, we define that when $Client$ sends a message to HLF , $Client$ cannot create a message and send it to itself. Similarly, in the $DenyServer2Server$ constraint, we specify that a HLF node cannot send a message to itself.

```
restriction DenyClient2Client:
"
All Client HLF key nonce #i. (
  Client2HLF(Client,HLF,<key,nonce>) @ #i
  & not(Client = HLF)
) ==> not (Ex #j nonce2 .
Client2HLF(Client,Client,<key,nonce2>) @j)
"
restriction DenyServer2Server:
"
All Client HLF key nonce #i. (
  HLF2Client(HLF,Client,<key,nonce>) @ #i
  & not(Client = HLF)
) ==> not (Ex #j nonce2 .
HLF2Client(HLF,HLF,<key,nonce2>) @j)
"
```

For the $dual_clients$ restriction, we specify that there do not exist two $Clients$ with the same key .

```
restriction dual_clients:
"
All Client key #i. (
  ID_Client(Client,key) @ #i
) ==> not (Ex #j Client2 .
ID_Client(Client2,key) @j)
"
```

The first lemma defines the secrecy of the key. For each $Client$, HLF , Key , two $nonces$ and two moments i and j in the situation: when there is a connection of $Client$ to HLF at moment i and HLF to $Client$ at moment j and moment j is after i and $Client$ cannot also be HLF then there is no such moment k that there is a revealed Key at moment k .

```
lemma Key_Secrecy:
"
All Client HLF Key nonce nonce2 #i #j. (
  Client2HLF(Client,HLF,<Key,nonce>) @ #i &
  HLF2Client(HLF,Client,<Key,nonce2>) @ #j &
  #i < #j &
  not (Client = HLF)
) ==> not (Ex #k1 . K(Key) @ #k1)
"
```

In the second lemma, we check the freshness. For each client $Client$ and node HLF and key t and two moments i and j in the situation: when there is a connection from $Client$ to HLF at moment i and HLF to $Client$ at moment j and moment j is after i where there are:

- 1) There is no such user *Client2* at moment *i1* that there is communication from *Client2* to *HLF* with the same key *KeyCH* at moment *i1* when *i* is equal to *i1*;
- 2) There is no user *HLF2* at moment *j1* that there is communication from *HLF2* to *Client* with the same key *KeyHC* at moment *j1* when *j* is equal to *j1*.

```

lemma freshness :
"
All Client HLF KeyCH KeyHC #i #j . (
Client2HLF(Client,HLF,KeyCH) @ #i &
HLF2Client(HLF,Client,KeyHC) @ #j &
#i < #j &
not(Client = HLF)
)==> (not (Ex Client2 #i1 .
Client2HLF(Client2,HLF,KeyCH)
@i1 & not (#i1 = #i)) & not (Ex HLF2 #j1 .
HLF2Client(HLF2,Client,KeyHC)
@j1 & not (#j1 = #j)))
"

```

For *Client* authentication, you need to prove that *Client* has a key that is used to protect the data before sending them. To do this, we have defined that for any *Client*, *HLF*, *nonce*, and *nonce2*, at moments *i* and *j*, there is a communication between *Client* and *HLF* where *HLF* sends a response to the request of *Client* and there is always a moment *k* such that *Client* has the *key* that was used to secure the communication before starting that communication.

```

lemma auth_Client :
"
All Client HLF key nonce nonce2 #i #j . (
Client2HLF(Client,HLF,<key,nonce>) @ #i &
HLF2Client(HLF,Client,<key,nonce2>) @ #j &
#i < #j &
not(Client = HLF)
)==> Ex #k . ID_Client(Client,key)
@k & #k < #i
"

```

The same is true for *HLF* authentication; it must also have *key* before communication can begin where *key* is used.

```

lemma auth_HLF :
"
All Client HLF key nonce nonce2 #i #j . (
Client2HLF(Client,HLF,<key,nonce>) @ #i &
HLF2Client(HLF,Client,<key,nonce2>) @ #j &
#i < #j &
not(Client = HLF)
)==> Ex #k . ID_Server(HLF,key)@k & #k < #i
"

```

In the Tamarin prover, we used the same simplifications as in Verifpal:

- 1) The key is a generated value, not a value generated from the parameters of *devic*;
- 2) The parameters themselves are also a generated value and not a list with numbers;
- 3) There is no way to compare timestamps.

As with Verifpal, Tamarin prover also confirmed key secrecy, freshness, and authentication of the *Client* and the *HLF* node during communication of the IoT device with the Hyperledger Fabric node. We also verified secrecy, authentication, and message freshness for the other operations.

In both cases, the result was positive. Both tools enforced identical constraints on us. Using the two tools allowed us to confirm that the protocol we developed is secure and provides the required security features. Using two tools reduced the probability that our proposed protocol did not meet our criteria because both tools use different deduction mechanisms.

VI. CONCLUSIONS AND FUTURE WORK

We have performed a formal modeling of the LAKEPFI protocol and presented the results of the analysis of its security properties. For our modeling and analysis, we have used two complementary formal verification tools, Verifpal and Tamarin. By choosing tools that use significantly different protocol models, we tried to minimize the possibility of erroneous verification. Based on the models developed using these tools, we verified the security properties of the protocol, such as message secrecy, authentication, and freshness. The choice of tools used to verify the protocol was not straightforward. We did not find a tool that was able to completely model our protocol. After testing many tools, we chose two that were closest to meeting our requirements. The difficulty in modeling our protocol lies mainly in generating the key to authenticate communication.

In the case of both tools, successful modeling of LAKEPFI required introducing some specific assumptions and simplifications. For example, the key is a randomly generated value; not a value generated explicitly from the device parameters. We have previously checked the randomness property of the created keys and, therefore, consider this assumption valid. Another limitation is that it is impossible to generate an array and randomly select the values from which the key should be created. Therefore, the array indexes are sent explicitly, which may affect the verification results. The final limitation is the inability to compare timestamps. In Verifpal, the possibility of evaluating values of timestamps is limited to the (unused) ASSERT function, while in Tamarin, one can define which events follow each other in time. However, one cannot compare the variables that store timestamps. The result of the time comparison influences the result of the execution of the protocol. If their difference is too significant, the message is considered invalid. Moreover, both tools that we used verify symbolic models. A symbolic model abstracts away the details of cryptographic operations and does not consider all implementation details, which could be seen as its limitation.

As part of future work, we plan to verify selected security properties of the LAKEPFI protocol in the computational model and validate the correctness and security of our implementation of the protocol. We also plan to test the performance of the protocol by experimenting with different use cases and configurations of the federated IoT environment.

ACKNOWLEDGMENT

This work has been partially funded by the NATO Allied Command Transformation Innovation Programme of Work and by the SEMACITI project, sponsored by the Ministry

of Defense of Republic of Poland as part of the Kościuszko Programme.

REFERENCES

- [1] N. Suri et al. “Exploring Smart City IoT for Disaster Recovery Operations”. In: *Internet of Things (WF-IoT), 2018 IEEE 4th World Forum on*. IEEE, 2018, pp. 463–468.
- [2] P. K. Panda and S. Chattopadhyay. “A secure mutual authentication protocol for IoT environment”. In: *Journal of Reliable Intelligent Environments* 6.2 (June 2020), pp. 79–94.
- [3] Z. Qikun et al. “Multidomain security authentication for the Internet of things”. In: *Concurrency and Computation: Practice and Experience* ().
- [4] U. Khalid et al. “A decentralized lightweight blockchain-based authentication mechanism for IoT systems”. In: *Cluster Computing* 23.3 (2020).
- [5] G. Shaoyong et al. “Master-slave chain based trusted cross-domain authentication mechanism in IoT”. In: *J of Network and Computer Applications* 172 (2020).
- [6] C. Chen et al. “A secure blockchain-based group key agreement protocol for IoT”. In: *The Journal of Supercomputing* (Feb. 2021).
- [7] M. Santos et al. “FLAT: Federated lightweight authentication for the Internet of Things”. In: *Ad Hoc Networks* 107 (2020).
- [8] M. Alshahrani and I. Traore. “Secure mutual authentication and automated access control for IoT smart home using cumulative Keyed-hash chain”. In: *J of Inf Sec and App* 45 (2019), pp. 156–175.
- [9] N. Kshetri. “Can Blockchain Strengthen the Internet of Things?” In: *IT Professional* 19.4 (2017), pp. 68–72.
- [10] N. Haur et al. *Building decentralized applications with Hyperledger Fabric and Composer*. Packt Publishing, 2018.
- [11] A. Babaei and G. Schiele. “Physical Unclonable Functions in the Internet of Things: State of the Art and Open Challenges”. In: *Sensors* 19.14 (2019).
- [12] K. Hofer-Schmitz and B. Stojanović. “Towards formal verification of IoT protocols: A Review”. In: *Computer Networks* 174 (2020), p. 107233.
- [13] C. B. Z. Shelby K. Hartke. *The Constrained Application Protocol (CoAP)*. Request for Comments (RFC) 7252. IETF, 2014.
- [14] C. Bormann and P. Hoffman. *Concise Binary Object Representation (CBOR)*. RFC 7049. IETF, Oct. 2013.
- [15] C. Bormann, M. Ersue, and A. Keranen. *Terminology for Constrained-Node Networks*. Request for Comments (RFC) 7228. IETF, May 2014.
- [16] B. Blanchet. “Security Protocol Verification: Symbolic and Computational Models”. In: *Principles of Security and Trust*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 3–29.
- [17] D. Dolev and A. Yao. “On the security of public key protocols”. In: *IEEE Transactions on Information Theory* 29.2 (1983), pp. 198–208.
- [18] A. Armando et al. “The AVISPA Tool for the Automated Validation of Internet Security Protocols and Applications”. In: *Computer Aided Verification*. Ed. by K. Etessami and S. K. Rajamani. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 281–285. ISBN: 978-3-540-31686-2.
- [19] N. Kobeissi. *Verifpal User Manual*. Manual. Symbolic Software, 2021. URL: <https://verifpal.com/res/pdf/manual.pdf>.
- [20] B. Blanchet et al. *ProVerif 2.04: Automatic Cryptographic Protocol Verifier, User Manual and Tutorial*. 2021.
- [21] D. Basin et al. “Symbolically Analyzing Security Protocols Using Tamarin”. In: *ACM SIGLOG News* 4.4 (2017).
- [22] M. Abadi. *Security Protocols and their Properties*. 2001.
- [23] H. Krawczyk. “The Order of Encryption and Authentication for Protecting Communications (or: How Secure Is SSL?)” In: *Proceedings of the 21st Annual International Cryptology Conference on Advances in Cryptology*. CRYPTO ’01. Berlin, Heidelberg: Springer-Verlag, 2001, pp. 310–331.
- [24] A. Zanatta. *Comparison of Tools for the Verification of Cryptographic Protocols*. 2021. URL: <https://github.com/AlessandroZanatta/Verification-of-Cryptographic-Protocols>.

Secure Onboarding and Key Management in Federated IoT Environments

Krzysztof Kanciak

Cybernetics Faculty

Military University of Technology

Warsaw, Poland

krzysztof.kanciak@wat.edu.pl

Konrad Wrona

NATO Cyber Security Centre /

Military University of Technology

The Hague, Netherlands / Warsaw, Poland

konrad.wrona@[ncia.nato.int,wat.edu.pl]

Michał Jarosz

Cybernetics Faculty

Military University of Technology

Warsaw, Poland

michal.jarosz@wat.edu.pl

Abstract—Many high-impact Internet of Things (IoT) scenarios, such as humanitarian assistance and disaster relief, public safety, and military operations, require the establishment of a secure federated IoT environment. One of the critical challenges in the implementation of federated IoT solutions involves establishing a secure and authenticated key management mechanism. We propose and validate in a laboratory environment a novel federated IoT onboarding and key management solution. Our dl-mOT protocol integrates an efficient identity-based modified Okamoto-Tanaka (mOT) protocol with a distributed ledger in order to establish an anchor of trust between federation members.

I. INTRODUCTION

MANY high-impact Internet of Things (IoT) scenarios, such as Humanitarian Assistance and Disaster Relief (HADR), public safety, and military operations, require the establishment of a secure federated IoT environment. Such a federation may involve entities with limited a priori trust relationships and generally rely on the integration and reuse of resources belonging to individual partners.

As an example, we consider a natural disaster, such as an earthquake, tornado, or flood, that affects a smart city. In such a scenario, military forces can be requested to assist local government agencies in delivering essentials, such as food and medical supplies, as well as medical and search and rescue support. One of the important operational priorities is to increase the number of information sources available to improve Situational Awareness (SA). To optimize the efficiency and effectiveness of their efforts, first responders can rely on data retrieved from the surviving smart city infrastructure. Such infrastructure may comprise sensors, actuators, and communication equipment, for example, traffic light posts with cameras for traffic flow monitoring, pollution and weather sensors, smart transportation networks, and smart power grids. To augment these capabilities, which can be degraded by a disaster, military forces can also deploy their own sensors at key locations, the data from which can be shared with local authorities and civilian responders and used to establish a common SA [1].

II. FEDERATED IOT ENVIRONMENT

An example of a federated IoT environment that was proposed for use within military and HADR operations is

presented in Fig. 1.

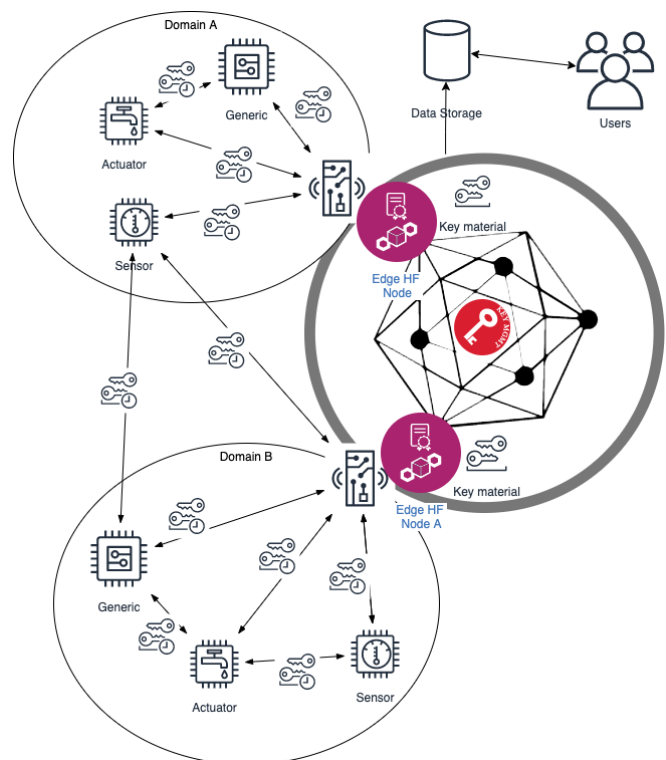


Fig. 1. Blockchain-based key management solution for IoT.

We can identify four types of components of such a federated IoT system:

- 1) IoT devices: Sensors and actuators, belonging to and operated by a specific organization and thus constituting a single security domain.
- 2) Edge nodes: These are gateway or sink nodes, facilitating communication within a single security domain and between devices belonging to different security domains. Edge nodes can also function as distributed ledger nodes.
- 3) Distributed ledger nodes: These are nodes participating in a permissioned distributed ledger. They represent different organizations participating in the federation. In the case of organizations operating an own IoT system, an

edge node can also play the role of a distributed ledger node. In the presented application, DL is responsible for authentication, authorization and for sharing the key for communication with IoT devices with each other. The ledger stores all the information necessary to perform the above operations. Each data saving transaction must be carried out only after fulfilling the conditions specified in the smart contract. In the case of Hyperledger Fabric, the smart contract is called chaincode. Because we use Hyperledger Fabric in our work, we will also rely on the naming convention used there.

- 4) End-user services: These are the primary consumers of sensor data and actuation capability offered by the IoT devices. Interaction between services and IoT devices is mediated via the federated distributed ledger. End-user services expose capabilities offered by a federated IoT system to the end-users.

In such an environment, IoT devices can communicate directly with each other, as well as with the edge nodes. A specific organization owns each IoT device, but to provide the required resilience and effectiveness of operation, it is desirable that once a federation is established, a device can communicate with any edge node belonging to the federation. Furthermore, a federation-wide federated access control policy can be maintained to define authorized direct communication patterns between IoT devices within and between organizations.

During the operational phases, IoT devices send data and requests to the edge node that acts as a mediator between IoT devices and distributed ledger nodes. Authorization to read and write data to the ledger is obtained by execution of a smart contract. The smart contracts can also be used to perform some processing of the data, e.g., data filtering, aggregation, or labeling.

III. CHALLENGES

The most critical challenges related to the interconnection of civilian and military communications and information systems (CIS) are related to security. In particular, the federated CIS needs to support controlled and timely information sharing between federation partners, meeting both stringent need-to-know constraints defined by the military partners, as well as responsibility-to-share requirements of effective execution of joint emergency response activities. Ensuring interconnection of military systems, usually compliant with specific military standards such as NATO Standardization Agreements (STANAGs), with their civilian counterparts, largely relying on open or commercial standards, introduces another layer of complexity and specific challenges. However, these interoperability issues are likely to decrease in the future due to an increasing focus on dual use of many of the new technologies, such as 5G, and an increasing focus of the military on cost-effectiveness of CIS implementation through the use of commercial-of-the-shelf (COTS) technologies and related civilian and open standards.

Federated emergency response and disaster recovery operations can be conducted in a broad spectrum of settings: in form of a peacetime support to civilian authorities as well as response to terrorist activities and humanitarian disasters in conflict regions. Therefore, in context of the CIS security, we need to consider presence of a powerful and technically sophisticated adversary, interested in disrupting response or using it as an opportunity to infiltrate or damage CIS of military partners. Such an attacker can compromise an arbitrary number of devices belonging to a specific federated organization. In the context of resilience to attacks, we differentiate between two types of devices. First type consists of devices equipped with a secure element (or a trusted processing module) that can provide reasonable resistance to access to secret data stored in the device, so that it can be assumed that for duration of a specific military operation, the data remains secret. Second type of devices is not equipped with any hardware security mechanisms - therefore, once compromised, any secret data stored at the device can be read and modified by the attacker. We assume that although attacker can compromise an arbitrary number of IoT devices, edge nodes and distributed ledger nodes belonging to specific federated organization, the attacker cannot compromise majority of the federated organizations. In particular, we assume that an attacker is not able to control majority of the distributed ledger nodes, representing different federated organizations, and thus a secure consensus and secure execution of distributed application (or so-called chain code) within distributed ledger is possible. Furthermore, we assume that although an attacker can inject malicious input data from the compromised nodes but all data is verified at the distributed ledger by means of smart contracts before it is written to the ledger and also before it is read.

IV. SECURITY REQUIREMENTS

The security mechanism should satisfy several specific requirements of the federated CIMIC. In federated operations, exact trust relationships between federation members might not be known in advance of device initialization. Therefore, a device might need to be re-authorized for operation within the specific federated environment and reconfigured to enable end-to-end encrypted information sharing between a sensor or actuator of one nation and a command and control system of another nation. The proposed mechanisms must be able to function in an adversarial environment, where static or cloud infrastructure might not be accessible for prolonged periods of time. To support IoT devices, security mechanisms must be efficient with respect to computing power and memory required. Similarly, they must scale well to scenarios that involve hundreds of devices and adapt to different communication patterns. For extremely constrained devices or when modification of device configuration is not possible, the use of an edge node or a digital twin for security adaptation should be considered in order to ensure secure integration with data consumer applications. Finally, we also need to ensure some typical security requirements. Perfect forward secrecy stipulates that compromise of a session secret shall not affect

the security of any previous or future sessions. Furthermore, compromise of any device and a master secret stored on the device should not affect the security of the system as a whole and the security of other devices. The system design should also support a change of cryptographic mechanisms, providing the ability to withstand new and emerging threats, such as quantum computing.

One of the critical challenges when implementing federated IoT solutions consists of bootstrapping and maintaining the system's security. To ensure the confidentiality and integrity of communication channels between IoT devices, a secure and authenticated key management mechanism must be implemented. Such a key management mechanism needs to fulfill several specific requirements:

- 1) Federated operations: The mechanism shall be compatible with federated operations, where the exact trust relationships between federation members might not be known in advance of device initialization. Therefore, there might be a need for a device to be re-authorized for operation within the specific federated environment.
- 2) Resilience and fault tolerance: The mechanism should function also in adversary environment, where some of static or cloud infrastructure is not directly accessible from an IoT device.
- 3) Support for constrained devices: The part of the key management mechanisms executed at the end-devices should be computationally efficient with respect to required computing power and memory.
- 4) Scalability: the mechanism needs to scale well to scenarios involving potentially hundreds of devices and adapt to different communication patterns, including both fully distributed scenarios and edge scenarios where end-devices communicate to a gateway connected to a backbone network.
- 5) Perfect forward secrecy: compromise of a session secret does not affect the security of any previous and future sessions
- 6) Robustness against compromise of individual devices: compromise of any device and of a master secret stored in the device should not affect the security of a system as a whole and security of other devices. This implies in particular that any potential key generation and device authentication authority needs to be implemented using a threshold-based approach.
- 7) Cryptographic agility: The system should support the change of cryptographic mechanisms, providing the ability to withstand new and emerging threats, such as quantum computing.
- 8) Increased security guarantees for key material: none of the federated organizations is able to obtain the cryptosystem master key.

V. DL-MOT: DISTRIBUTED LEDGER IMPLEMENTATION OF THE MODIFIED OKAMOTO-TANAKA PROTOCOL

To meet the challenges defined in Section III, we propose a novel key management solution for IoT devices that integrates

with a permissioned distributed ledger. The integration of the distributed ledger is a novel element that provides increased robustness against attacks on key generation and authentication authority.

In an IoT system, there are substantial differences between the computational capabilities of various devices. Therefore, our design aims at optimizing computational overhead induced on constrained IoT devices in exchange for much higher overhead on the edge nodes side, that are significantly more powerful.

The proposed solution is based on the Okamoto-Tanaka identity-based key management protocol, initially proposed in [2] and further extended in [3] and [4]. We further modify the protocol by implementing a distributed key generation authority based on a distributed ledger. Integration of a distributed ledger into identity-based cryptographic solutions can be seen as a more general and reusable security pattern; its applications in the context of privacy protection have been discussed in [5].

Since the concept of identity-based public-key cryptography was introduced in [6], many identity-based authenticated key exchange protocols have been proposed [7]–[9]. Most of the proposed protocols rely on elliptic curve pairings that are computationally very expensive and hard to implement in constrained devices. In the context of an IoT, a one-round ID-based key exchange with forward secrecy over the Rivest-Shamir-Adelman algorithm (RSA) group is a much more compelling choice. Moreover, although in data-centric federated systems, attribute- or identity-based encryption can be used to enable an efficient and effective enforcement of access control policies through data encryption, there is an inherent problem with pairing-based cryptography: There is no way to perform pairing operations (to generate private keys) in a decentralized and accountable manner. Therefore, it is difficult or impossible to maintain the principles of zero-trust architecture in the system.

When it is possible to securely store key material on an IoT device, e.g. using a Trusted Processing Module (TPM), we propose using *dl-mOT*, an enhanced version of the modified Okamoto-Tanaka (mOT) protocol [3]. The mOT protocol enables two parties to use their identities to establish their common secret keys without sending and verifying public key certificates. We further enhance the mOT protocol by integrating it with a distributed ledger. In this way, we address some inherent weaknesses related to the use of identity-based cryptography. In particular, we mitigate key, escrow, remove a single point of failure, and add strong accountability for key generation events.

The advantage of the dl-mOT protocol is its computational and communication efficiency and ease of implementation. The protocol can be implemented with short exponents, e.g., 160-bit exponents with a 1024 bit modulus. Moreover, operating over an RSA group enables the implementation of multiparty computation protocols, such as distributed exponentiation and multiparty generation of an RSA modulus.

Another essential feature of the dl-mOT protocol is the

support for perfect forward secrecy (PFS) [10] that was identified as one of the key security requirements in our federated environment. Perfect forward secrecy means that the leakage of a long-term key used by an entity does not compromise the security of session keys established by that entity. Moreover, dl-mOT allows spontaneous decentralized device-to-device interactions, without any request to the edge node or distributed ledger. Edge nodes are only involved in secure onboarding of the sensor into the federation. Device bootstrapping is a subprocess of onboarding and requires just enough information exchange between a device and the network to establish a secure channel.

A. Onboarding

In the mOT protocol, a Key Generation Center (KGC) chooses the RSA parameters $N = pq$, exponents d, e , and a random generator g of the cyclic subgroup of quadratic residues QR_N . The parameters p, q, d , and e are random and safe primes, that is, a prime p is safe iff $\frac{p-1}{2}$ is also a prime. KGC publishes values N, e, g , as well as two hash functions H and H' as a set of public parameters P . During the onboarding phase, every device receives from the KGC a unique identity id_D and a corresponding private key $S_D = H(id_D)^d \bmod N$. It is important to note that the mOT software requires the storage of secure private keys on the sensors, which is not always possible.

B. Operational phase — Point-to-point communication scenario

Communicating devices are already onboard, which means that they have their identities and secrets $S_A = H(id_A)^{d_i} \bmod N_i$. Now, a dynamic asynchronous secure channel may be established. In the key agreement phase, devices A and B choose ephemeral private exponents x and y , respectively. Bar notation denotes single domain keys, and hats distinguish multi-domain keys establishment. Each device can calculate

$$\begin{array}{ccc} A & \xrightarrow{\alpha = g^x S_A \bmod N} & B \\ A & \xleftarrow{\beta = g^y S_B \bmod N} & B \end{array}$$

Fig. 2. mOT key agreement

the mOT session key in the following way:

$$\bar{K}_A = (\beta^e / H(id_B))^{2x} \bmod N, \quad (1)$$

$$\bar{K}_B = (\alpha^e / H(id_A))^{2y} \bmod N. \quad (2)$$

Both values are equal since:

$$\begin{aligned} \bar{K}_A &= (\beta^e / H(id_B))^{2x} = \\ &= \left((g^y H(id_B)^d)^e / H(id_B) \right)^{2x} = \\ &= (g^{ey} H(id_B)^{ed} / H(id_B))^{2x} = \\ &= (g^{ey} H(id_B)^1 / H(id_B))^{2x} = g^{2xye} = \bar{K}. \end{aligned} \quad (3)$$

Analogically, $\bar{K}_B = \bar{K}$. The mOT session key K is established with a key derivation function H' :

$$K = H'(\bar{K}, id_A, id_B, \alpha, \beta). \quad (4)$$

C. Multi-domain KGC setting

In our setting, each IoT device belongs to an organization that operates its own security domain and the KGC. In a federated IoT environment, devices from different organizations, and thus belonging to different security domains, should be able to execute the key agreement protocol and communicate.

Each organization i operates its own KGC_i with public parameters P_i , as previously described, and secret key d_i , such that $e_i d_i = 1 \bmod \phi(N_i)$.

The device A belonging to the organization operating KGC_i receives during onboarding a secret $S_A = H(id_A)^{d_i} \bmod N_i$. The identity of the device consists of $H(id_A)$ and P_i . Similarly, the device B belonging to the organization that operates KGC_j has a secret $S_B = H(id_B)^{d_j} \bmod N_j$ with public parameters P_j . Consider

$$E = \text{lcm}(e_1, e_2) \quad (5)$$

$$\hat{g} = (g_i \bmod N_i, g_j \bmod N_j) \quad (6)$$

$$\hat{N}_i = (N_i^{E/e_i} \bmod N_i, 1 \bmod N_j) \quad (7)$$

$$\hat{N}_j = (1 \bmod N_i, N_j^{E/e_j} \bmod N_j) \quad (8)$$

Both communicating sensors compute a pair of values $\bmod N_i N_j$:

$$\hat{S}_A = (S_A \bmod N_i, 1 \bmod N_j) \quad (9)$$

$$\hat{S}_B = (1 \bmod N_i, S_B \bmod N_j) \quad (10)$$

respectively. Device A chooses an ephemeral random integer x and computes

$$\hat{X} = \hat{g}^x \bmod N_1 N_2 \quad (11)$$

and sends to B value

$$\hat{\alpha} = \hat{X} \hat{S}_A \bmod N_1 N_2. \quad (12)$$

The device B chooses a random secret y and sends to A the value:

$$\hat{\beta} = \hat{Y} \hat{S}_B \quad (13)$$

where

$$\hat{Y} = \hat{g}^y. \quad (14)$$

Finally, A computes the shared secret \hat{K} :

$$\begin{aligned} \hat{K} &= \left(\frac{\hat{\beta}^E}{\hat{B}} \right)^{2x} = \left(\frac{\hat{Y}^E \hat{S}_B^E}{\hat{B}} \right)^{2x} = \hat{g}^{2xyE} \left(\frac{\hat{S}_B^E}{\hat{B}} \right)^{2x} = \\ &= \hat{g}^{2xyE} \bmod N_1 N_2. \end{aligned} \quad (15)$$

The device B performs an analogous computation. Similarly as within a single organization, at the end, both devices make a hash of the shared secret and the identities of both parties:

$$K = H'(\hat{K}, id_A, id_B, \hat{\alpha}, \hat{\beta}). \quad (16)$$

The boot-strapping process (considered as a subprocess of onboarding) can be repeated during the lifetime of a device and requires direct communication to the edge node and identity registration in the distributed ledger.

D. Multiparty private exponentiation

In the scenarios presented, the KGCs are the only parties that own the master secrets of the cryptosystems. When onboarding an IoT device within a domain, an exponentiation using a secret exponent known to the KGC is required. With the help of a bit-decomposition [11], we have constructed a constant-round protocol for multiparty private exponentiation where several KGCs participate in secret key generation (since attaching a new device implies private key generation which is single integer exponentiation). Multiparty exponentiation in the IoT device setup phase eliminates the single point of storing master secrets and supports zero-trust architecture principles. We assume that federated organizations share the domain master secret and perform their part of the exponentiation process to achieve strong accountability of the onboarding process and to weaken requirements for trust into edge nodes. The bit-decomposition private exponentiation is computationally expensive and a significant amount of research has been devoted to improve its performance [12], [13]. However, in our scenario, it is performed by relatively powerful KGC nodes and only during the onboarding of an IoT device. The classical approach to distributed modular exponentiation on arithmetic circuits relies on bit-decomposition, which was first proposed in [11]. To achieve an exponentiation protocol with a public base b and secret exponent e , the authors of [11] propose a method for securely bit-decomposing the inputs (bd) secretly shared l -bit exponent e into bits, using the so-called fan-in multiplications. We assume that there exists a function:

$$[e]_{bits}bd([e]) \quad (17)$$

that receives a secret shared input $[e]$ and returns its shared bit-decomposition $[e]_{bits}$, so that it produces l shares:

$$([a_0], [a_1], \dots, [a_{l-1}]) \quad (18)$$

where $a_i \in \{0, 1\}$. The final result of the exponentiation is then the product of the decomposed multiplications:

$$\prod_{i=0}^{l-1} ([a_i]b^{2^i} + [1] - [a_i]). \quad (19)$$

The identity of an IoT device is public, while its private key is generated using KGC secrets and multiparty private exponentiation. The objective is to improve the auditability of the onboarding process (or bootstrapping, understood as a subprocess of onboarding) and make it impossible for a single compromised or malicious KGC node to add devices to the domain. This was achieved through integration of bit-decomposition with a distributed ledger, ensuring that every decomposed multiplication, performed in support of private exponentiation, left a trail on the distributed ledger. It means that federated organizations share their KGC secrets, and

each execution of multiparty exponentiation is registered in distributed ledger so that it is known among federation which organizations took part in the sensors onboarding process.

E. Use of trusted execution environment

In our initial proof-of-concept KGCs master-secrets escrow problem was solved by encapsulation into Intel Software Guard Extensions (SGX) enclave. Intel SGX is an implementation of the concept of a trusted execution environment in Intel CPUs that allows users to define secure enclaves. Secure enclaves are regions of memory whose content is protected and unable to be read or saved by any process outside the enclave itself. Combining chaincode with SGX makes it possible to run applications that demand privacy, such as distributed exponentiation combiners, which was first proposed in [14]. In our initial design, a KGC enclosed in a secure enclave acted as an oracle for distributed ledger chaincode, and there was no risk of keys compromise, but still the enclaves where only parties where keys were stored and enclave availability was necessary. But only the master-secrets distribution and multiparty exponentiation weakened the nodes availability requirement sufficiently, so that zero-trust architecture principles can be satisfied.

Intel SGX comes with two advantages. The first is that enclaves provide confidentiality and integrity of any code and data inside the enclave. The second advantage of enclaves is a mechanism that allows remote parties to verify the identity of the enclaves via a process called attestation. Intel SGX provides applications with the ability to create areas in memory called enclaves that provide confidentiality and integrity for the code/data running inside it, even in the presence of buggy or malicious privileged software or actor. Attestation is a process that allows remote parties to verify the identity of a piece of software. The remote party can use Intel's SGX attestation hosted service to verify the quote before provisioning any secrets into the enclave. Intel SGX also supports cryptographic binding of secrets to an enclave, the so-called *sealing*. Any enclave can use a hardware-generated key, called a sealed key, to encrypt the secrets with the key. The key is available only to the owner enclave running on the same platform. This is crucial for key material management.

VI. PERFORMANCE DISCUSSION

An important practical question is the prediction of the performance of the dl-mOT in various configurations of a federated IoT environment. The main aspects of performance analysis are related to the overheads introduced by onboarding, key management, and recording of transactions in the distributed ledger.

A. Onboarding overhead

dl-mOT protocol requires a secure channel or controlled environment for the onboarding phase. During device onboarding, an RSA group element, that is, a 1024-bit private key, and a hash result, that is, a 256-bit value, must be transmitted. These values (in total 160 bytes) must be stored securely on

the sensor device. There is no computational overhead for the sensor during the onboarding phase. However, there is a computational cost induced by a device onboarding on the KGC side. In the single-domain scenario, this computational overhead comprises one exponentiation and two hash operations. In a distributed multiparty scenario, the computational cost is significant, as described in V-D, but all computations occur on nodes without significant computational restrictions.

B. Key management overhead

The communication overhead of the operational phase of dl-mOT is as in the original Diffie-Hellman protocol and requires two messages with a single RSA group element. After these messages have been exchanged, the symmetric key with full Perfect Forward Secrecy against active attackers is established and remains confidential to all parties, including the federated organization and KGC. Furthermore, due to the identity-based properties of mOT, there is no communication overhead related to certificate transmission that is typical for the authenticated Diffie-Hellman (DH) key exchange protocol. The computational cost of key establishment using the dl-mOT protocol, when used with short exponents, is very low. dl-mOT is more efficient than any RSA-based key agreement protocol and any authenticated DH protocol over Z_p^* for large prime and much more efficient than any of the ID-based protocols based on elliptic curve pairings. The dl-mOT protocol is less efficient than protocols using elliptic curves, but only for keys longer than 2048 bits.

C. Distributed ledger overhead

In the dl-mOT scenario considered, a distributed ledger is used to generate identity-based keys for IoT devices, which can be further used for secure communication with the distributed ledger nodes and the IoT peer nodes. Our target environment consists of a distributed ledger based on Hyperledger Fabric. Furthermore, we assume the use of a policy that specifies that any two organizations are sufficient to sign the transaction. Our prediction is that more nodes result in shorter delay times; that is, the median delay related to the generation of private keys and the recording of public key parameters in the distributed ledger is shorter for more nodes.

The performance of the dl-mOT solution directly depends on the performance and scalability of the distributed ledger, and in the case of our design on Hyperledger Fabric performance, which acts as a trustworthy storage for validated and reliable data.

When writing data to the ledger, the number of requests (or transactions) that can be processed increases with the number of nodes that participate in the ledger from each organization. However, the transaction processing time will also increase as the number of federated organizations that need to accept (sign) a transaction increases. To counteract the performance penalty introduced with the increasing number of federated organizations, an appropriate endorsement policy can be used in Hyperledger Fabric. The endorsement policy determines how many organizations are needed to approve the transaction.

If the endorsement policy requires fewer organizations to accept a transaction, the ledger performance and, therefore, the performance of dl-mOT will increase.

When reading data from a ledger, no consensus among organizations is required, and therefore, the performance depends only on the number of available ledger nodes; i.e., the more nodes are available, the higher performance of the ledger.

VII. RELATED WORK

The interoperability aspects of civilian IoT platforms have been discussed in several earlier papers. A high-level architecture for semantic and syntactic interoperability between cloud-based IoT platforms has been proposed in [15]. A multiauthority access control framework for federated cloud IoT platforms has been presented [16]. This earlier work differs significantly from our contribution and is not directly applicable to military and civil-military scenarios. First, we consider the federation and interoperability of IoT systems in multiple layers. In HADR and military operations, we cannot rely on universal availability of cloud connectivity, but we also target disadvantaged and adversary environments, which may rely on locally deployed IoT enclaves and edge nodes interconnected via private and proprietary links. Moreover, most of the earlier work does not consider explicitly resilience and survivability aspects - the distribution is interpreted rather in context of preserving control, than providing reliability and fault tolerance. In our solution, we focus on ensuring that during the operational phase, i.e. after onboarding, the IoT devices and end users can be authenticated and authorized by any other node belonging to federation, thus mitigating some of the risks related to defects, environmental faults, and adversarial activity.

Several ongoing activities focus on the development of secure and interoperable onboarding and configuration management mechanisms for IoT devices [17], [18]. Our proposed key management protocols are aligned with the frameworks presented in [19] and [20].

The use of blockchains in the context of IoT applications was a topic of several recent surveys, for example, [21]–[23]. However, in the context of our solution, the most relevant work is related to key and access management, as well as the performance evaluation of blockchain-based applications.

The use of blockchain to implement a key management approach in a federated environment with smart vehicles was investigated in [24]. A pairwise key management scheme based on a transitory system-wide secret has been discussed in [25]. The use of randomness obtained from IoT devices for the generation of cryptographic material has been mainly investigated in the context of taking advantage of Physical Unclonable Functions (PUF) [26], [27]. An excellent study of various sources of secure PUFs is provided in [28].

The performance of blockchain, when used to store data obtained from IoT devices, has been studied in [29]. Similarly, the performance of Solidity smart contracts implemented in Ethereum for the management of access to IoT devices is investigated in [30]. This investigation was further extended

in [31], where a comparison of the performance of blockchain-based access management with an alternative CoAP-based solution is presented. The applicability of the Ethereum smart contract for access control in the IoT is also discussed in [32].

We also chose to provide identity-based authenticated key exchange in our device network as a solution for environments where the operation of traditional public key infrastructure (including its various phases of a life cycle, such as certificates dissemination and revocation) is too costly in terms of bandwidth, on-device storage, and computation. Since it is possible to store additional key material in an IoT device, we propose using a modified Okamoto-Tanaka (OT) protocol [2] adapted to the decentralized setting. Our solution is based on [3], which introduced a protocol called mOT. The OT protocol enables two parties to use their identities to establish their common secret keys without sending and verifying public key certificates.

As noted in [3] in terms of computational effort, the OT protocol is more efficient than any RSA-based key agreement protocol and any authenticated Diffie-Hellman protocol over Z_p^* for large prime. OT protocol is incomparably more efficient than any of the ID-based protocols based on elliptic curve pairings. OT is pretty close in terms of computational efficiency to the certificate-based MQV protocol, which runs over elliptic curves, while mOT runs over RSA composites. The cost of MQV on-line is slightly larger than that of OT, but OT requires a setup phase. For moderate security parameters (1024-bit RSA), OT has a computational advantage over MQV, but for larger modulus (above 2000 bits), the advantage is on the elliptic-curve side.

Current key management protocols [15], [16] do not offer such a combination of properties.

We aim to provide a two-message identity-based key exchange, maintaining low computational and communication overhead, achieving the PFS property, and allowing sensors to communicate without interaction with edge nodes. Since the federated IoT environment includes trusted edge nodes, our protocol also allows the implementation of a multiauthority case in which IoT devices from two separate domains (connected to a single edge node) can establish a secret key. In such a case, each edge node plays the role of an independent key generation center (since it belongs to a trusted network or at least a semi-trusted network).

OT framework can also be used as an enabler to implement the Self-Sovereign Identities (SSI). The objective of SSI is to provide subjects with complete control of their own digital identities [33]. There are two standard approaches to SSI, namely, Decentralized Identifiers (DID) and Verifiable Credentials (VC) [34] - both of them rely on public-key cryptography. To control a specific DID, a subject just has to own a private key associated with a public key in the DID document. Although DID focuses on cryptographic identification, VC provides authenticated and privacy-aware attribute disclosure. The mentioned SSI approaches mean that devices need to be able to execute encryption algorithms based on asymmetric keys, which can be challenging in devices with

limited processing and energy resources, and cope with the communication overhead of transmitting metadata, such as DID and VC. Furthermore, in the IoT world, low communication overhead plays a vital role, as wireless communication protocols often offer relatively small packet sizes. In particular, low-energy protocols such as Long Range Radio (LoRa) and Bluetooth Low Energy (BLE) have maximum packet sizes of 222 and 244 bytes, respectively, and the mentioned approaches require at least 512 bytes or more, which means that additional mechanisms are required, for example, for message partitioning and lost packet control.

Therefore, although SSI may improve security and privacy protection for IoT devices, new, more efficient cryptographic methods need to be identified to ensure its successful deployment in the IoT environment. In particular, power and communication constraints of small devices in large distributed networks suggest the benefits of using a one-round authenticated key exchange (AKE) protocol in the IoT environment. Our proposed approach revisits the classic Okamoto-Tanaka protocol [3] that realizes single-round key agreement and exchange with perfect forward secrecy, using a single group element per message and maintaining low computational and communication overhead, avoiding the need to distribute large public key certificates.

VIII. CONCLUSIONS AND FUTURE WORK

We have presented dl-mOT, a novel key management solution for federated IoT environments. Our solution integrates an efficient identity-based mOT protocol with a distributed ledger. It relies on a distributed ledger to establish an anchor of trust between federation members.

Our work focused only on a subset of challenges related to the implementation of an operational federated IoT environment. In particular, we have not discussed how the resources available in the federation can be discovered or how to achieve secure time synchronization. We plan to investigate the possible integration of our key management approach with other open frameworks, such as Tesseract E4¹, to implement a solution that can be validated in practice. Moreover, public-key algorithms that we use in our current implementation of the distributed ledger are susceptible to quantum computing attacks. Although it is possible to modify Hyperledger Fabric to use customized public-key cryptography, the integration of quantum-safe algorithms into our solution is left for future work.

ACKNOWLEDGMENT

This work has been partially funded by the NATO Allied Command Transformation Innovation Programme of Work and by the SEMACITI project, sponsored by the Ministry of Defense of Republic of Poland as part of the Kościuszko Programme.

¹<https://teserakt.io/>

REFERENCES

- [1] F. T. Johnsen, Z. Zieliński, K. Wrona, N. Suri, C. Fuchs, M. Pradhan, J. Furtak, B. Vasilache, V. Pellegrini, M. Dyk, M. Marks, and M. Krzysztoń, "Application of iot in military operations in a smart city," in *2018 International Conference on Military Communications and Information Systems (ICMCIS)*, May 2018, pp. 1–8.
- [2] E. Okamoto and K. Tanaka, "Key distribution system based on identification information," *IEEE Journal on Selected Areas in Communications*, vol. 7, no. 4, pp. 481–485, May 1989.
- [3] R. Gennaro, H. Krawczyk, and T. Rabin, "Okamoto-Tanaka revisited: Fully authenticated Diffie-Hellman with minimal overhead," in *Proc. of Applied Cryptography and Network Security (ACNS)*, vol. 6123 LNCS, 2010, pp. 309–328.
- [4] B. Tian, F. Wei, and C. Ma, "mOT+: An efficient and secure identity-based diffie-hellman protocol over RSA group," in *INTRUST 2014: Revised Selected Papers of the 6th International Conference on Trusted Systems*, vol. 9473, 2015, pp. 407–421.
- [5] K. Kanciak and K. Wrona, "Towards an Auditable Cryptographic Access Control to High-value Sensitive Data," *Int. J. Electron. Telecommun.*, vol. 66, no. 3, pp. 449–458, 2020.
- [6] A. Shamir, "Identity-Based Cryptosystems and Signature Schemes," in *Proc. of the Annual Int. Cryptology Conf. (Crypto)*, 1984.
- [7] A. Kate and I. Goldberg, "Distributed Private-Key Generators for Identity-based Cryptography," in *Int. Conf. Secur. Cryptogr. Networks*, 2010.
- [8] X. Boyen and B. Waters, "Anonymous hierarchical identity-based encryption (Without random oracles)," in *Adv. Cryptol. - CRYPTO*, 2006.
- [9] D. Boneh and M. Franklin, "Identity-Based Encryption from the Weil Pairing," *SIAM J. Comput.*, vol. 32, no. 3, pp. 586–615, 2003.
- [10] R. Canetti and H. Krawczyk, "Analysis of Key-Exchange Protocols and Their Use for Building Secure Channels," *Cryptology ePrint Archive*, Report 2001/040, 2001, available at: <https://eprint.iacr.org/2001/040>.
- [11] I. Damgård, M. Fitz, E. Kiltz, J. B. Nielsen, and T. Toft, "Unconditionally Secure Constant-Rounds Multi-party Computation for Equality, Comparison, Bits and Exponentiation," in *Theory of Cryptography*, S. Halevi and T. Rabin, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 285–304.
- [12] C. Ning and Q. Xu, "Constant-rounds, linear multi-party computation for exponentiation and modulo reduction with perfect security," in *Advances in Cryptology - ASIACRYPT 2011*, D. H. Lee and X. Wang, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 572–589.
- [13] —, "Multiparty computation for modulo reduction without bit-decomposition and a generalization to bit-decomposition," in *Advances in Cryptology - ASIACRYPT 2010*, M. Abe, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 483–500.
- [14] M. Brandenburger, C. Cachin, R. Kapitza, and A. Sorniotti, "Blockchain and Trusted Computing: Problems, Pitfalls, and a Solution for Hyperledger Fabric," arxiv, 2018. [Online]. Available: <http://arxiv.org/abs/1805.08541>
- [15] I. P. Zarko, S. Mueller, M. Plociennik, T. Rajtar, M. Jacoby, M. Pardi, G. Insolubile, V. Glykantzis, A. Antonic, M. Kusek, and S. Soursos, "The symbIoTe solution for semantic and syntactic interoperability of cloud-based IoT platforms," in *Global IoT Summit, GloTS 2019 - Proceedings*. Aarhus, Denmark: IEEE, 2019, pp. 1–6.
- [16] S. Sciancalepore, G. Piro, D. Caldarola, G. Boggia, and G. Bianchi, "On the Design of a Decentralized and Multiauthority Access Control Scheme in Federated and Cloud-Assisted Cyber-Physical Systems," *IEEE Internet of Things J.*, vol. 5, no. 6, pp. 5190–5204, 2018.
- [17] S. Symington, W. Polk, and M. Souppaya, "Trusted Internet of Things (IoT) Device Network-Layer Onboarding and Lifecycle Management," NIST, Working Paper, 2020.
- [18] M. Sethi, B. Sarikaya, and D. Garcia-Carrillo, "Secure IoT Bootstrapping: A Survey," IETF, Internet Draft, 2020.
- [19] M. Vucinic, G. Selander, J. Mattsson, and D. Garcia, "Requirements for a Lightweight AKE for OSCORE," IETF, Internet Draft, 2020.
- [20] F. Palombini, L. Seitz, G. Selander, and M. Gunnarsson, "OSCORE Profile of the Authentication and Authorization for Constrained Environments Framework," IETF, Internet Draft, 2020.
- [21] M. A. Ferrag, M. Derdour, M. Mukherjee, A. Derhab, L. Maglaras, and H. Janicke, "Blockchain technologies for the internet of things: Research issues and challenges," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2188–2204, 2019.
- [22] M. Wu, K. Wang, X. Cai, S. Guo, M. Guo, and C. Rong, "A Comprehensive Survey of Blockchain: From Theory to IoT Applications and beyond," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8114–8154, 2019.
- [23] W. Viriyasitavat, L. D. Xu, Z. Bi, and D. Hoonsopon, "Blockchain Technology for Applications in Internet of Things - Mapping from System Design Perspective," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8155–8168, 2019.
- [24] A. Lei, H. Cruickshank, Y. Cao, P. Asuquo, C. P. Ogah, and Z. Sun, "Blockchain-Based Dynamic Key Management for Heterogeneous Intelligent Transportation Systems," *IEEE Internet of Things Journal*, vol. 4, no. 6, pp. 1832–1843, 2017.
- [25] F. Gandino, R. Ferrero, B. Montrucchio, and M. Rebaudengo, "Fast Hierarchical Key Management Scheme with Transitory Master Key for Wireless Sensor Networks," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 1334–1345, 2016.
- [26] B. Chen and F. M. Willems, "Secret Key Generation over Biased Physical Unclonable Functions with Polar Codes," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 435–445, 2019.
- [27] P. Gope and B. Sikdar, "Lightweight and Privacy-Preserving Two-Factor Authentication Scheme for IoT Devices," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 580–589, 2019.
- [28] NATO STO IST-ET-104, "Physical Unclonable Functions (PUFs) in Military IoT," NATO STO, Tech. Rep., 2019.
- [29] M. Alaslani, F. Nawab, and B. Shihada, "Blockchain in IoT Systems: End-to-End Delay Evaluation," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8332–8344, 2019.
- [30] O. Novo, "Blockchain Meets IoT: An Architecture for Scalable Access Management in IoT," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 1184–1195, 2018.
- [31] —, "Scalable access management in IoT using blockchain: A performance evaluation," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4694–4701, 2019.
- [32] Y. Zhang, S. Kasahara, Y. Shen, X. Jiang, and J. Wan, "Smart contract-based access control for the Internet of Things," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1594–1605, 2019.
- [33] G. Fedrecheski, J. Rabaey, L. Costa, P. Ccori, W. Pereira, and M. Zuffo, "Self-Sovereign Identity for IoT environments: A Perspective," in *Global Internet of Things Summit (GIoTS)*, 2020.
- [34] M. Sporny, D. Longley, and D. Chadwick, "Verifiable credentials data model 1.0," W3C, Tech. Rep., 2019, <https://www.w3.org/TR/2019/REC-vc-data-model-20191119/>.

Anomaly detection on compressed data in resource-constrained smart water meters

Sarah Klein

Sirris, Brussels, Belgium

Email: sarah.klein@sirris.be

Anna Hristoskova

Sirris, Brussels, Belgium

Annanda Rath

Sirris, Brussels, Belgium

Renaud Gonce

Shayp, Brussels, Belgium

Abstract—Increasing amount of devices in our daily life are equipped with sensors that transfer information to a cloud solution where the data is finally analysed. By improving the data intelligence on the edge, the data transfer can be reduced, which not only saves bandwidth and thus reduces energy consumption, but also leads to increased privacy protection. In this paper, we propose a privacy-friendly water leakage detection approach for various kind of water meters (optical and digital) performed on a very constrained, wireless devices.

I. INTRODUCTION

THE amount of IoT devices on our homes increases for several years already. While most of them promise easier living and higher comfort, the actual applications are often neither environment nor privacy friendly. In this paper, we introduce a water leakage detection algorithm that takes both into account by running analytics directly on a long-living, energy-saving device in a privacy-preserving manner. For this, we apply a very lightweight data compression at the edge that enables leakage detection and significantly reduces the bandwidth needed to transfer the data from the edge to a central cloud device. By this, public as well as private buildings all over the world can profit from a solution that is easy to install and that reduces the waste of fresh water. This work was performed in close collaboration between Sirris and Shayp, who are leading specialists in water efficiency and monitoring, such that the evaluation of the approach was performed on real-world data.

The remainder of this paper is organized as follows: We will first provide an overview of current solutions in section II, before describing the technical setup in section III. In section IV we will explain in detail our solution for an on-the-edge leakage detection that respects privacy aspects. Its evaluation will be discussed in V-1 on both, artificial and real-world data before concluding in section VI.

II. RELATED WORK

Recently, several solutions on leakage detection by smart meters were suggested. The common approach entails a connected device at water meter level, which sends pulse data to a cloud back-end for further collection and analysis. Hence, the leakage detection is executed only in the cloud [1], [2], [3]. In [1], a solution for remote monitoring of water consumption and leakage detection is discussed. The setup consists of a water meter is connected to an Arduino micro-controller that

sends the data to a Raspberry Pi gateway. Only from there, the data is hourly transferred to a cloud back-end that the end user can consult. From a hardware point of view, this solution is error prone as the two-fold messaging can lead to data loss and increased latency. Further, a wall socket has to be available in the vicinity of the water meter which makes it hard to generalise the solution.

The leakage detection suggested by the authors is performed at cloud level and consists of the detection of four different scenarios: (i) a negative consumption trend, which indicates a problem with the data transfer; (ii) a continuous water flow over 24 hours, which indicates a strong leakage; (iii) a coincidence with the last two measurements, which also indicates a leakage; and finally (iv) a significant deviation from the historical consumption. However, this solution suffers from a cold-start problem for detecting leakages reliably. Further, some of the rules are only meaningful in residential buildings, as e.g. hospitals can indeed show a significant hourly water consumption throughout an entire day.

Similarly, in [3], a wireless open source middleware was developed. Also here, the data is collected locally via an edge gateway but the leakage detection is only performed at cloud level. The empiric algorithm looks for the absence of water consumption during a specific interval across the day that deviates from historical consumption. With this rather rough estimate, it is possible to detect leakages in residential buildings, but it will probably fail in other types of buildings like hospitals, which were not investigated in the publication.

The solution presented in [4] makes use of wireless, battery-driven vibration sensors, instead of directly measuring the water consumption. Increasing vibrations in a pump indicate a pipe burst in the network. The proposed system uses a lightweight edge anomaly detection algorithm based on compression rates. The leakage detection splits the data x from the stream into two equally long sequences of length w_{stream} , e.g. at time t , the data sequences consist of $[x(t - 2w_{stream}, \dots, x(t - w_{stream}))]$ and $[x(t - w_{stream}, \dots, x(w_{stream}))]$. For both, they apply miniLZO compression [5]. When the compression rate changes significantly from one window to the next, they assume that a leakage occurred. By using lossless compression in their lab-based test rig, the authors could reduce communication by 90% compared to periodical messaging which leads to an increasing battery lifetime. While this algorithm offers very

suitable performance for big pipes and big bursts, it does not detect small leakages that are most common in residential households or schools [6]. Further, the authors do not consider privacy or security aspects critical when reducing the amount of messages sent. In case of a residential building, the number of messages being sent can already leak private information on the consumption pattern of this particular household. Lack of messages poses a more critical security thread as one can assume people are away.

III. TECHNICAL SETUP

Shayp's solution for monitoring water consumption and detecting anomalies in water usage consists of a wireless water meter reading device. It has long-range, battery-powered data logger compatible with all pulse-ready meters, specifically designed to withstand water immersion, harsh conditions and ensure an ideal performance in deep indoor situations. This is enabled using a Narrowband IoT (NB-IoT) connection to a back-end cloud. In addition of being low-power, this radio standard also offers a very good connectivity, allowing communication inside deep basements where water meters are usually to be found. Coupled to the periodic sending schema as explained below and thanks to the low-power micro-controller of the device, NB-IoT allows to meet a 10-year battery lifetime.

Through a pulse emitter placed on the water meter, the water consumption takes the form of pulses, each of which, depending on the water meter, corresponds to a certain amount of water (typically 1L or 10L) consumed per defined time window. A sensor connected to the pulse emitter detects these pulses. The data logger collects these pulses and sends this consumption data to a cloud back-end where water consumption analysis and leak detection take place.

1) *Periodic sending schema*: Shayp's device aggregates hourly consumption data in periods of 30 s and then sends a message to the cloud. Although the message length is 512 bytes, the actual used space is about 152 bytes (32 bytes for payload and 120 bytes for 120 data points). This results in 360 unused bytes in a message.

2) *Cloud solution for leakage detection*: Currently, Shayp supports leakage detection at the cloud level. For residential buildings this takes between 1-3 hours, while corporate buildings vary from 3 up to 24 hours. This is due to the complex water consumption pattern for buildings such as schools and hospitals, where water usage is constant. The lack of on-device analytics and the hourly message pattern prevent any anomaly detection in less than an hour. The data made available for this research consists of the number of consumption pulses and the estimated leakage size in pulses per 30 seconds. Further, information about the type of building, e.g. a public building or a private household, is provided.

With the method described in this publication, we improve the detection time even further (<1h for residential, <3h for corporate) by applying a lightweight analysis executed on the device at near real-time. This should however not hamper the lifetime of the battery, which ideally could even be extended

to up to 16 years, which is the expected life time of water meters.

IV. LEAKAGE DETECTION

In this work, we introduce a leakage detection algorithm operating on a very constrained edge device based on data compression. The algorithm has to fulfill the following requirements in order to be applicable in real-world scenarios:

- 1) Minimal power consumption in order to guarantee at least 16 years of battery lifetime.
- 2) Fast leakage detection such that a warning can be sent with short delay (<1h for residential, <3h for corporate buildings).
- 3) Preserve the privacy of the underlying data

Interestingly, the three points above contradict each other: i.e. in order to increase the battery lifetime, the device should send as few messages as possible. But with less messages, it is harder to detect leakages early enough. One solution could be to only send messages once water consumption was measured and a risk of leakage was detected. Though, under this assumption, privacy and security in households or corporate buildings are at risk, as in this case an attacker can easily derive consumption patterns by profiling the message sending patterns.

In order to deal with these contradicting requirements, we developed a leakage risk assessment at the edge based on the compression of the consumption data. This approach is combined with a leakage-sensitive random messaging schema in order to ensure privacy preservation.

1) *Data Consumption Compression*: Per message, 480 bytes of data can be sent. In order to use this space as efficiently as possible, while not losing information, we apply a lightweight lossless sequential compression. We either use:

- 1) Fibonacci codes [7] on the unprocessed sequence, as it has proven to be very robust and efficient [8].
- 2) A combination of run-length encoding (RLE) [9] and Fibonacci codes where in a first step we apply RLE and only after Fibonacci codes.

For each device, more than 200.000 data points were used. One can see that the overall performance of the combined compression (RLE + Fibonacci) leads to better results, hence higher compression rates, for most devices (Fig. IV-1). Nevertheless, two extreme outliers can easily be detected, i.e. the two devices at the bottom rows, namely *NE83A580* and *N389F2E8*. By analysing the statistics of the consumption and leakage pulses given in Table I, it is possible to explain why the compression is so different in these cases: For (*N389F2E8*), the overall consumption is high such that the raw Fibonacci encoding is less effective as the strings encoding the integers are becoming longer. The mean leakage pulses for this device is NaN, as no leakage was detected by the leakage detection algorithm that is used in production. For *NE83A580*, the difference in compression between the sole Fibonacci encoding and combined compression indicates that the consumption values are fluctuating a lot such that the hardly any longer periods of

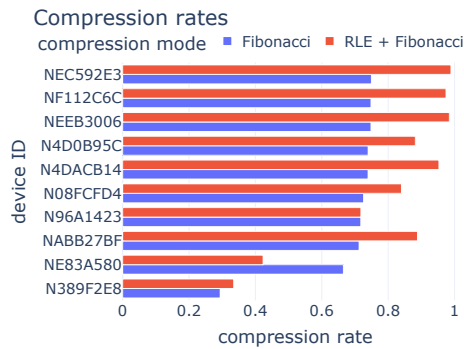


Fig. 1. Compression ratio for all devices for the two different compression approaches.

TABLE I
OVERALL STATISTICS OF CONSUMPTION AND LEAKAGE PULSES

device ID	consumption pulses	leak pulses
NEC592E3	0.004244	0.000005
NF112C6C	0.015171	0.004336
NEEB3006	0.016815	0.000212
N4D0B95C	0.085269	0.017191
N4DACB14	0.113889	0.000000
N08FCFD4	0.226471	0.045114
N96A1423	0.308077	0.113322
NABB27BF	0.475513	0.027130
NE83A580	0.784921	NaN
N389F2E8	21.086333	0.210611

the same (or no) consumption are given in the data. This gives a hint for (continuous) leakage in the data as also indicated by the high number of mean leakage pulses given in Table I. We will in more detail discuss the results per device in subsection V-2.

2) *Anomaly detection*: In most production-ready solutions, leakages in water supply are detected at cloud level [3], [1] where sufficient computational resources and historical consumption data are available. The approach we present here aims at estimating the risk of a leakage at edge level as early as possible. For this, we use the specific consumption patterns of a leaking asset. In case of a leakage, the consumption pulses $c(t)$ of a specific strength occur very regularly - the consumption of a leakage. Mathematically speaking, we define $\delta = |F(c(t_m))| |RLF(c(t_m))|^{-1}$, where $|F(c(t_m))|$ is the length of the Fibonacci code of the consumption sequence $c(t_m)$ during a time window t_m , and $|RLF(c(t_m))|$ is the length of the combined compression (RLE + Fibonacci encoding) of the same sequence. Based on δ , we define the binary variable $L(c(t_m))$ indicating whether a leakage was detected in sequence $c(t_m)$ as being 1 for $\delta > \epsilon$ and 0 otherwise, where ϵ , the leakage threshold, is a free parameter of the model. This inflation given by a high δ -value can be used for leakage detection. In the case of a comparably strong leakage, the combination of RLE and Fibonacci codes is about twice as long as the sequences encoded by Fibonacci codes.

The overall idea is similar to the one in [4] but while they compare the compression in two consecutive windows, we

compare the compression of the data within the same window with two different compression algorithms. This does not only reduce the possible alarm delay as our window can be shorter, it also takes into account the specific consumption pattern in case of a leakage.

3) *Random sending schema*: The sending schema is crucial in order to fulfill the requirements listed in the beginning of Section IV. When sending as few messages as possible the battery lifetime is strongly extended. The simplest solution is thus sending a message once the message space of 480 bytes is filled or a specific anomaly is detected. However, water consumption in residential buildings, similar to electricity consumption [10], is highly privacy sensitive. Even if we assume that the messages are properly and strongly encrypted, an attacker can deduce information on the consumption pattern and the presence of inhabitants by only counting the number of messages per time interval. Though a schema in which messages are sent in regular intervals prevents this possible privacy breach, it should be avoided: In order to increase the battery lifetime, the regular sending interval should be as large as possible but this leads to a significantly delayed leakage detection. For this reason, we suggest a privacy-preserving sending schema based on random timing as sketched, described in detail below. The algorithm takes the following variables as input:

- T : the expectation time for sending a message,
- ϵ : The leakage threshold as defined above
- ω : the look-back window on which to calculate the leakage risk, e.g. 2 hours
- n_{\max} : the maximal warnings to send per detected leakage
- R : the *force* radius
- ζ : a constant that defines noise in the sending pattern.
- b : The maximal sending time, e.g. 24 hours.

In a first step, we draw a random sending time interval t_i for $i \in \mathbb{N}_0$ from a truncated exponential distribution $f_T(t|\lambda, b)$ for $0 < t \leq b$ where $\lambda = \frac{1}{T} > 0$ and save it to the list of *random times*. We use a truncated function in order to ensure that messages are sent within time b (e.g. 24h). Every time a new sensor measurement is available (in our case every 30s), we run the following steps:

- 1) Calculate the encoding values of the sequence in the look-back window $F(c(\tau_j^i))$ and $RLF(c(\tau_j^i))$, where τ_j^i is the time of the j -th measurement in the i -th message.
- 2) If $\tau_j^i = t_i$, hence the time when the message should be sent, send the message with content (leakage risk, encoded sequence) and then calculate the average of the last R sending times from the *random times* list. We use this average sending time T^* in order to calculate the next random sending time t_{i+1} drawn from the distribution

$$f(t_{i+1}|\lambda, b) = \frac{\frac{1}{\lambda} \exp\left(-\frac{t}{\lambda}\right)}{1 - \exp\left(-\frac{b}{\lambda}\right)} \text{ with } \lambda = T + K \frac{T - T^*}{T^*}. \quad (1)$$

We can see this similarly to a canonical description of a confined ideal gas in an external potential [11], where

T is the equilibrium value and the potential is given by the earlier sending of a message due to a detected leakage. The system is pushed back to equilibrium with the artificial temperature K . We then add the calculated next sending time t_{i+1} to the list of random sending times and set $i = i + 1$ and $j = 0$.

- 3) Otherwise, if $\tau_j^i \neq t_i$, calculate δ from $F(c(\tau_j^i))$ and $RLF(c(\tau_j^i))$. In case $\delta > \epsilon$, hence $L(c(t_m)) = 1$ and the number of sent warnings $n < n_{\max}$, draw a normally distributed random number $r = \mathcal{N}(\zeta, 0.2\zeta)$. In case $\tau_j^i < (n + 1)r$, send the message. We use this in order to add an additional level of randomness to the sending pattern as well as to prevent too many warnings for the same detected and continuous leakage. The factor $n + 1$ gives here an additional damping. Just as above, calculate the new random sending time as given in eq. (1) and add it to the list of sending times.
- 4) In all other cases, wait for the next measurement.

In the next section, we will evaluate our approach on artificial as well as real-world data. We will judge our method by the accuracy of the detected leakages, the number of sent messages, which indicates the battery consumption on the device and will further perform an analysis on the distribution of sending times.

V. EVALUATION

In this section, we will evaluate the proposed leakage detection and messaging algorithm on the edge with respect to the leakage detection accuracy and privacy preservation. In order to perform the analysis of the leakage detection performance, we will not only use the real-world consumption data but also artificially created data as in that case the actual starting point of the leakage is exactly known and we can perform exact statistics on leakage detection time. For privacy evaluation, we will use the real-world data only.

1) *Artificial data*: In order to create the artificial data, we extract the daily consumption per weekday from the data without leakages. For this, we first manually remove the consumption pulses related to leakages from the time series data from device *N96A1423*, which is a school building. Hence, the general consumption patterns for weekdays are very different from those on weekends but also the overall consumption per day is quite irregular. For each day of one year of artificial data, we select every 30 seconds randomly a consumption value from the same time and day of week from the historical consumption. We add some normally distributed noise from $r = \mathcal{N}(0, 0.2)$ and round the resulting value to an integer. In this way, the overall consumption pattern per day of the week is kept. Based on this consumption without leakages, we add pulses of leakages of different severity (S), different strength (s) and different length (Δt) to create the final artificial consumption. The severity S defines how strong the leakage is, i.e. if $S = n > 1$ at every $1/n$ -th pulse, s pulses are added to the normal consumption for the following measurement during Δt .

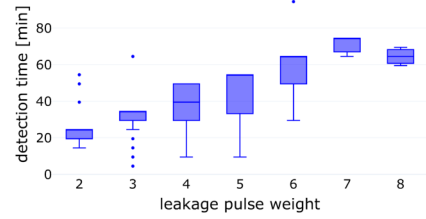


Fig. 2. Time until detection for different pulse severity and strength calculated on artificial data.

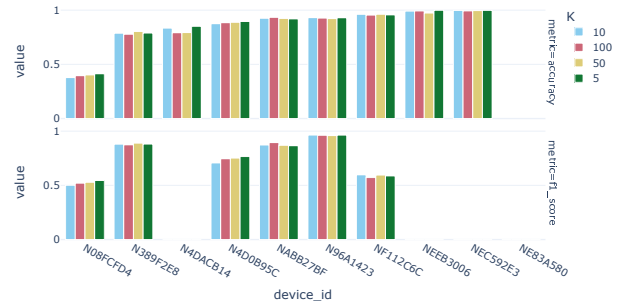


Fig. 3. Accuracy (top) and F1-score (bottom) of the leakage detection on the edge for all devices compared to the cloud solution. The color of the bars indicates the artificial temperature K .

With an overall accuracy of more than 93% and a F1-score of 90% over all samples, our edge algorithm performs very well even on 30 s data granularity. Further, the detection time (Fig. 2) on the edge ranges from less than 30 min from leakages with $S = 2$ to about one hour for leakages with lower severity. This fulfills the requirements of leakage detection of 1-3 hours as defined above.

2) *Real-world data*: For the evaluation of the leakage detection performance on the real-world data, we use the cloud-based leakage detection as ground truth. We expect our accuracy to be slightly lower, as we might detect leakages earlier. Hence, the ground truth does not yet indicate a leakage

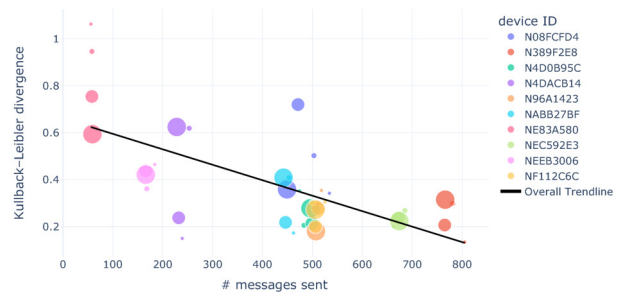


Fig. 4. Kullback-Leibler divergence for all devices given by the different colors. The size of the markers indicates the artificial temperature K .

though it has already started. In Fig. 3, we show the accuracy and the F1-score. For 80% of devices, the accuracy is higher than 0.8, while for one of the remaining devices (*NE83A580*), the ground truth data is missing. Further, the F1-score, the ratio of correctly detected instances of leakages, are similarly high as the accuracy. Note that the devices with a F1-score of zero are those devices that have (quasi) no leakages (c.f. Table I). For both devices (*NEC592E3*, *NEEB3006*), we only see a leakage in the first 5 minutes of the full data set which rather indicates a misclassification due to a cold start problem. The devices with lower F1-score, namely *N08FCFD4* and *NF112C6C*, are a school and a sports facility, respectively. By analysing the data in more detail, the following factors lead to a reduced accuracy and F1-score: i.e. first, for *N08FCFD4*, the average leakage is at 0.049 pulses, hence every twentieth pulse indicates a leakage. With the current threshold, this leakage is too weak to be detected reliably. However, for such a weak leakage, which is probably a dripping water tap, the timely detection is less crucial. For the latter, device *NF112C6C*, we can see that at the time when the actual leakage begins, the leakage detection at the edge triggers a warning almost 24 hours earlier than the cloud algorithm, which explains the reduced and in this case misleading F1-score.

Additionally, in order to ensure that it is not possible to extract privacy-sensitive information from the number of message that are sent, we calculate the Kullback-Leibler (KL) divergence $D_{KL}(P||Q)$ [12]. We chose the KL divergence as it is usually used in adversarial Neyman–Pearson tests for identifying distribution differences [13]. For P , we derive the discrete distribution from a sample of random variables drawn from the truncated exponential distribution as given in equation (1) derived from as many samples as messages were sent (for each device and temperature) with $T^* = T$. Q is the discrete distribution derived from the random sending times when applying our algorithm. We use it as a measure of information gained when approximating the truncated exponential distribution P with the out-of-equilibrium distribution Q that enables early leakage warnings.

In Fig. 4 the divergence is shown against the number of send messages for all devices (color) and for different artificial temperatures (size). There is no clear correlation between the artificial temperature and the KL-divergence and hence does not change anything in the information that can be extracted from the distribution. On the contrary, the KL is clearly related to the number of messages being sent. Hence, over long run times of the algorithm, the KL divergence decreases such that no private information can be extracted from the distribution of messages sending times.

VI. CONCLUSION AND NEXT STEPS

We introduced an on-the-edge water leakage detection approach that addresses three contradicting requirements: (i) an overall reduced number of messages in order to extend the device’s battery lifetime, (ii) early-alarmed in case of a detected leakage on the edge, and (iii) a privacy-ensuring message sending schema. The approach is based on lightweight

compression performed on the edge in order to timely detect leakages and on random messaging for privacy protection. We evaluated it against artificial as well as against real-world data from devices installed in different types of buildings, such as private households and public buildings. In both cases, we observe a high leakage detection accuracy as well as a timely detection of the leakage, fulfilling the industrial requirements.

As next steps, we plan to further improve our approach by considering building-type specific leakage detection. Additionally, we will analyse our approach on pipe bursts, which show a very specific pattern. From our initial results, we see already that we receive reliable warnings also for those. Further, we will perform an on-the-edge battery lifetime study in order to give a realistic estimation when applying our approach in production.

ACKNOWLEDGMENT

The work in this paper results from MIRAI, a project labelled by ITEA3 under project no 19034, with funding support from Innoviris Brussels, Belgium.

REFERENCES

- [1] H. Fuentes and D. Mauricio, “Smart water consumption measurement system for houses using iot and cloud computing,” *Environmental Monitoring and Assessment*, vol. 192, no. 9, pp. 1–16, 2020. doi: 10.1007/s10661-020-08535-4
- [2] M. Fagiani, S. Squartini, L. Gabrielli, M. Severini, and F. Piazza, “A statistical framework for automatic leakage detection in smart water and gas grids,” *Energies*, vol. 9, no. 9, p. 665, 2016. doi: 10.3390/en9090665
- [3] S. Alvisi, F. Casellato, M. Franchini, M. Govoni, C. Luciani, F. Poltronieri, G. Riberto, C. Stefanelli, and M. Tortonesi, “Wireless middleware solutions for smart water metering,” *Sensors*, vol. 19, no. 8, p. 1853, 2019. doi: 10.3390/s19081853
- [4] S. Kartakis, W. Yu, R. Akhavan, and J. A. McCann, “Adaptive edge analytics for distributed networked control of water systems,” in *2016 IEEE First International Conference on Internet-of-Things Design and Implementation (IoTDI)*. IEEE, 2016. doi: 10.1109/IoTDI.2015.34 pp. 72–82.
- [5] J. Kraus and V. Bubla, “Optimal methods for data storage in performance measuring and monitoring devices,” in *Proceedings of electronic power engineering conference*, 01 2008. ISBN 9788021436503 pp. 131–133.
- [6] T. Britton, G. Cole, R. Stewart, and D. Wiskar, “Remote diagnosis of leakage in residential households,” *Journal of Australian Water Association*, vol. 35, no. 6, pp. 89–93, 2008.
- [7] A. Apostolico and A. Fraenkel, “Robust transmission of unbounded strings using fibonacci representations,” *IEEE Transactions on Information Theory*, vol. 33, no. 2, pp. 238–245, 1987. doi: 10.1109/TIT.1987.1057284
- [8] S. T. Klein and M. K. Ben-Nissan, “On the usefulness of fibonacci compression codes,” *The Computer Journal*, vol. 53, no. 6, pp. 701–716, 2010. doi: 10.1093/comjnl/bxp046
- [9] A. Robinson and C. Cherry, “Results of a prototype television bandwidth compression scheme,” *Proceedings of the IEEE*, vol. 55, no. 3, pp. 356–364, 1967. doi: 10.1109/PROC.1967.5493
- [10] C. Beckel, L. Sadamori, and S. Santini, “Automatic socio-economic classification of households using electricity consumption data,” in *Proceedings of the fourth international conference on Future energy systems*, 2013. doi: 10.1145/2487166.2487175 pp. 75–86.
- [11] C. Kittel, *Elementary statistical physics*. Courier Corporation, 2004.
- [12] S. Kullback and R. A. Leibler, “On information and sufficiency,” *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951. doi: 10.1214/aoms/1177729694. [Online]. Available: <http://www.jstor.org/stable/2236703>
- [13] Z. Li, T. J. Oechtering, and D. Gündüz, “Privacy against a hypothesis testing adversary,” *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 6, pp. 1567–1581, 2018. doi: 10.1109/TIFS.2018.2882343

Self-adaptive Device Management for the IoT Using Constraint Solving

Ghada Moualla, Sebastien Bolle, Marc Douet
Orange Labs
38 Meylan, France
Email: Firstname.Lastname@orange.com

Eric Rutten
Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LIG,
F-38000 Grenoble France,
Email: Eric.Rutten@inria.fr

Abstract—In the context of IoT (Internet of Things), Device Management (DM), i.e., remote administration of IoT devices, becomes essential to keep them connected, updated and secure, thus increasing their lifespan through firmware and configuration updates and security patches. Legacy DM solutions are adequate when dealing with home devices (such as Television set-top boxes) but need to be extended to adapt to new IoT requirements. Indeed, their manual operation by system administrators requires advanced knowledge and skills. Further, the static DM platform — a component above IoT platforms that offers advanced features such as campaign updates / massive operation management — is unable to scale and adapt to IoT dynamicity. To cope with this, this work, performed in an industrial context at Orange, proposes a self-adaptive architecture with runtime horizontal scaling of DM servers, with an autonomic Auto-Scaling Manager, integrating in the loop constraint programming for decision-making, validated with a meaningful industrial use-case.

I. INTRODUCTION

WITH Device Management (DM), an operator or a service provider is able to remotely manage connected devices deployed at the customer’s premises. The main DM features are [1]: (i) Provisioning: which targets device initial and in-life configuration, (ii) Monitoring: which allows detecting anomalies such as malfunctioning devices using log traces and data collection, (iii) Maintenance: which allows firmware and configuration updates, and (iv) Troubleshooting: which is remote actions to fix service and device errors.

Currently, DM solutions are widely deployed and mainly used for Smart Home service management. However, IoT platforms with DM features and standards have been developed to incorporate DM features such as firmware and configuration update, for two reasons : one is to accommodate the expansion of the Internet of Things (IoT) that offers a wide range of new smart applications [2] (e.g., Smart City, Smart Building, Smart Industry), the other is for environmental reasons of sustainability (e.g., device reuse, and lifespan enhancement).

While connectivity and cloud analytics are considered essential aspects of an IoT architecture, one of the most critical is the management of IoT [3]. This will help register connected devices, bring them online efficiently, ensure that they work properly and securely after being deployed, and send configuration or firmware updates remotely. However, with a very large number of IoT devices distributed across one or more geographic locations, monitoring and maintaining these devices can be an overwhelming task if done at the individual

physical level. Moreover, the conventional centralized DM approach becomes a serious limitation.

To face these limitations, this work is performed in an industrial context at Orange, and proposes an approach to self-adaptive Device Management for the IoT using constraint solving, validated on an industrial use-case. We introduce a new IoT DM architecture based on an autonomic manager, called Auto Scaling Manager (ASM). We have leveraged the MAPE-K (Monitor-Analyze-Plan-Execute over a shared Knowledge) Autonomic Computing reference architecture [4] to build this manager that is able to manage the DM system and adapt at runtime the required number of DM servers to handle the evolution of both the IoT device fleet and the physical infrastructure. Furthermore, a constraint programming model [5] is integrated into this manager and used for decision-making on the placement of DM service within the infrastructure.

The adaptive solutions for the scaling and placement of distributed components is a well-known topic in the context of distributed systems and the Constraint Programming (CP) has also used for a variety of real-world optimisation problems including placement problem. However, our main contribution involves a Constraint Programming-based autonomic loop approach in the context of DM IoT. In which, the system information is gathered and analyzed at runtime, in order to dynamically revise and adapt the constraint optimization criteria.

We evaluate experimentally the feasibility of our approach considering a privacy use-case. The objective is to analyze the ASM behavior in terms of adaptation decisions and to show how it scales horizontally, with respect to the evolution of both the DM system and the physical infrastructure. Our contributions are:

- An autonomic DM architecture for IoT devices to handle the device fleet evolution at runtime. For that, a Constraint Programming paradigm is integrated into an autonomic feedback loop.
- An experimental validation of the architecture for a privacy scenario.

The paper is structured as follows. Section II covers the background and the related work for the good understanding of our work. Section III states the targeted problem. Section IV details our proposed architecture. Section V details our

experimental evaluation and results. The last section concludes our paper and raises some future perspectives.

II. STATE OF THE ART

A. Device Management in IoT Context

DM operations are performed remotely via a management server, managed by a DM operator, that sends out the management operations to the management clients, hosted on the managed devices to ensure their proper functionality. The server and clients communication is based on dedicated protocols, e.g., TR-069 [6], OMA Lightweight Machine to Machine (*LWM2M*) [7].

DM of IoT is vital to maintain the proper functionality of IoT devices, keep them secure, and ensure the evolution and maintenance of their growing number, while anticipating new demands for devices and services. Compared to the legacy DM, new challenges are faced [8], [9]: *(i)* Heterogeneity: moving from a limited set of device types to a wide variety of devices [2], *(ii)* Dynamicity: moving from rather stable internal states, i.e., battery and network conditions, towards versatile states [10], *(iii)* Privacy: many IoT services leverage sensor data to adapt to the local context. It is critical to adapt these services to protect end users privacy and customers confidential and personal information [11], and *(iv)* Scalability: moving from a few devices managed by centralized systems to a massive number of devices that needs a distributed management.

In the works [12], [13], authors proposed: *(i)* automating of DM operations, and *(ii)* adjusting the execution speed of the campaign firmware update to device and infrastructure capabilities, i.e., hardware, current load and network congestion to address IoT heterogeneity and dynamism. Their adaptation strategy is based on operational measures, e.g., the error rate of DM operations and the infrastructure response time. Our work takes a further step towards scalability and privacy management by distributing DM operations close to the customer's premises based on Edge Computing.

Cloud computing consists in delivering computing and storage services over the Internet. It offers the advantages of flexibility and the ability to store/analyze data. However, when cloud computing is used for IoT, new challenges emerge.

With the huge number of heterogeneous IoT devices, IT operational bandwidth (BW) consumption is significant, especially from managing device provisioning, commissioning, decommissioning, and ensuring firmware upgrades. Further, as more devices appear, there is a need for a scalable DM solution to adapt to varied deployment scenarios and enable seamless integration and management of these distributed devices. We believe that Edge Computing, with all its features, has a promising potential to ensure end-users privacy (among other benefits as security and latency) when performing DM operations, compared to a centralized DM platform hosted in the cloud.

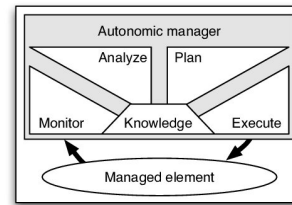


Fig. 1: MAPE-K Autonomic Loop, problem in reference [4]

B. Autonomic Computing

Autonomic Computing (AC) is defined as the self-management capabilities of a system [14] to respond to the increasing system administration complexity. It has proven to be effective in minimizing administrator involvement in the computer systems management, where an autonomic system is able to adapt to both external and internal changes by re-configuring itself (Fig. 1 shows the AC reference architecture).

Due to the huge number of heterogeneous devices involved in IoT system, the manual management and maintenance is impractical. In [15], the authors advocated that to address the problems of manual management, automatic approaches is needed. In their study, they stated why autonomic computing is useful and how to use it in the IoT context. Furthermore, automated architectures for IoT DM have been proposed in [13], [12] for automated device targeting (i.e., defining the appropriate devices for a DM operation) and for detecting errors/anomalies before generalizing patching to all target devices in a given fleet).

C. Service Placement in Fog and Edge using Constraints

Given the massive number of IoT devices, along with their related applications, the applications deployment in the Fog/Edge is needed to cope with the latency and privacy requirements [16]. This placement problem has been addressed in the literature [16], [17], [18] with several solutions based on different application scenarios, e.g., monolithic applications, network assumptions, e.g., infinite link bandwidth, and objective functions e.g., latency, cost. These solutions can be classified as follows: *(i)* Exact solution using Integer Linear Programming (ILP) [19]/ Constraint Programming [20], *(ii)* Approximations, and *(iii)* Heuristics [21] and Meta-heuristic, e.g., Genetic Algorithm (GA).

The placement solutions, that mainly rely on ILP or on heuristic approaches, are not generic enough to deal with the features of all applications [20], as they are: *(i)* uneasily extensible to incorporate new application / infrastructure features or to integrate new placement constraints and *(ii)* non-upgradable to exploit any user-implied resolution approach.

CP has these following appealing features: *(i)* It provides a generic and easy-to-upgrade service placement, *(ii)* It provides a faster solution even in a reasonably large scale environment (in [20] it is compared to and outperformed other algorithms such as ILP and GA with 1200 nodes), and *(iii)* The CP code is small and easy to implement. Thus, CP seems attractive to

adopt it for DM problem (More details in Sec. V). Many open-source CP solvers are available, such as the Choco solver [22] and OR-tools [23] and other commercial solvers, such as IBM CPLEX CP Optimizer [24].

For solving the Service Placement Problem (SPP), different optimization strategies are proposed in the literature [18], [16]. We distinguish two categories: (i) mono-objective that optimizes only one objective function, and (ii) multi-objective optimization that optimizes simultaneously many objective functions. These optimization functions are: (i) Latency: for delay-sensitive applications, (ii) Resource utilization: such as minimizing used bandwidth, (iii) Cost: There are two types of costs: the networking cost for data transmission charges and associated expenses, and other expenses related to storage, migration, and so on, (iv) Energy consumption: it includes when the service is sent by the end-user to the Edge device, when the Edge node processes the service; and when the Edge needs the Cloud, and (v) Other metrics: such as congestion ratio, i.e., ratio between the service links requirements and the physical links capacity.

D. State of the Art Synthesis

Device management is a fundamental issue, especially when it comes to the IoT environment, with its emerging challenges. To cope with these challenges, Autonomic Computing with its self-management capabilities, appears as an attractive solution for managing the DM system. Further, leveraging the benefits of the Edge Computing paradigm, we aim to provide the DM operations for IoT clients in a distributed manner to meet both the clients' requirements and the providers' concerns.

To this end, relying on Edge Computing together with Autonomic Computing MAPE-K architecture, we provide an autonomic manager with the ability for autonomous horizontal scaling in terms of the number of DM servers to adapt to the evolution of both the DM device fleet and the physical infrastructure. Learning from the literature and moving a step further, we rely on Constraint Programming to model and solve at runtime the DM placement problem by integrating a CP solver into the autonomic loop of our proposed autonomic manager. Moreover, to the best of our knowledge, our work is the first work that tackle the DM of IoT devices in a distributed manner.

III. PROBLEM DEFINITION

Deploying and managing large fleet of IoT devices is challenging due to their geographical distribution [25] and it can be an overwhelming task if done at the individual physical level. Ensuring the up-to-date state of this fleet is perturbed by fleet composition variations or by the availability of new firmware or configuration. The variation depends on two types of events: Device arrival which corresponds to the first connection of a customer's newly acquired object that usually comes with an outdated factory-installed firmware; and Device departure that occurs when an owner unsubscribes to an operator's offer, requiring a configuration to reset the device

to factory settings or when a device shuts down their network access to save energy for instance.

As a result, there is a strong and growing need for an autonomous system to manage the device fleet. This feature can considerably reduce the time and effort required to sustain the health and the performance of devices throughout their lifecycle. The automation of the DM system was considered in works [12], [13] which gave us a starting point for our work.

Moreover, when it comes to firmware updates, e.g., security patches or other updates, called over-the-air (OTA) updates, the conventional centralized DM approach becomes a serious limitation since designing a DM system to handle hundreds of devices is entirely different from designing one to handle billions. The consumption of IT bandwidth resulting from managing these devices will be significant. Therefore, an elastic DM system is very essential to ease and ensure secure, fast and proper batch updates for device fleet while removing the pressure on both the nodes that host the DM components and the links that are used for server-clients communication by scaling well to manage this huge device fleet.

Knowing that, the Edge computing architecture, which brings essential data processing capabilities close to the network edge, provides a compelling solution for IoT DM use case that addresses the following issues.

- Save Bandwidth: DM operations do not have to be sent over long routes between the server hosted in a data center and the end devices. With Edge Computing, the DM operations could be provided through the distributed servers via different Edge nodes at different locations near to the end users;
- Guarantee the privacy: Management of IoT end devices locally (e.g., gateway at the client geographical place) or in the proximity of the end devices could ensure the user's privacy. Given this privacy need, a DM server can be restricted to be deployed on specific areas (zone), namely Dedicated Zones (DZs). For example, a factory requiring device operations to be performed inside its LAN (Local Area Network);
- Support devices mobility: Edge computing is better suited to support end-user mobility than the centralized DM platform and to enable the seamless firmware updates management while ensuring the required latency.

IV. PROPOSED DM ARCHITECTURE

The proposed architecture for self-adaptive IoT DM is depicted in Fig. 2. It allows the dynamic horizontal scaling of the distributed DM servers based on designated policies (e.g., physical resource utilization thresholds, Privacy requirement).

It has two main layers: (i) Control layer which is comprised of a centralized autonomic manager called ASM and a system administrator that represents an external manager-to force external actions when needed or to give high-level objectives - and (ii) Infrastructure layer that involves the physical nodes/links that hosts the Administrative layer, i.e., DM servers, and the IoT devices.

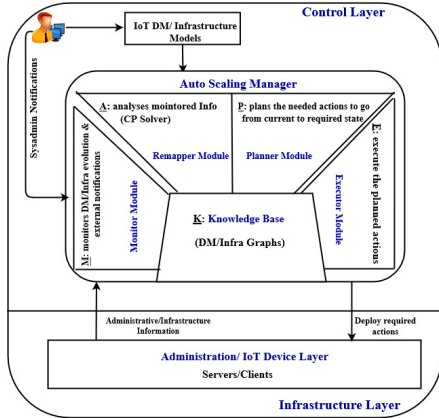


Fig. 2: Self-adaptive IoT DM Architecture based on MAPE-K reference

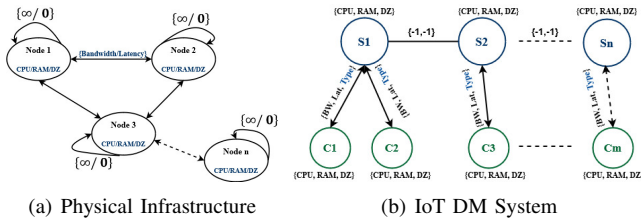


Fig. 3: Input Graph Models

This architecture is based on two main models, depicted as graphs, which represent the DM system and the underlying infrastructure (more details in Sec. V). On the one hand, the infrastructure model (Fig. 3(a)) is dedicated to the specification of the underlying topology that will host the DM service components. On the other hand, the DM system model (Fig. 3(b)) is intended to the representation of the DM system entities along with their requirements (i.e., servers and clients).

The lifecycle associated with this architecture involves these two models that will evolve over time, e.g., arrival/departure of DM clients, failures in infrastructure nodes/links. These models are used as the input of a specific constraint program integrated into the (Re)Mapper module of the ASM. In the Sec. IV-A, we explain ASM in more details.

A. Auto Scaling Manager (ASM)

This autonomic manager is built following the MAPE-K architecture and is responsible for managing of the DM system from its design till the end of the system's lifecycle. Thus, the objective is to specify a particular DM system /Infrastructure topology to be modeled, by a system administrator or DM service provider, and then handled at runtime. This implies making horizontal scaling decisions of the DM servers to adapt to any new condition dynamically, i.e., adding/removing one or more DM servers at runtime. This work is realized through the following modules:

- Monitor module: is responsible of observing the evolution of both DM system and infrastructure and forwarding the information to the Remapper module for analysis;
- (Re)Mapper module: analyses the information from the system administrator or the Monitor to decide on the

target state of the DM system and then it solves the DM system placement within infrastructure (See Sec. V) ;

- Planner module: Based on the Mapper information, it extracts and selects the required actions to move to the required state of DM system (e.g., add/remove DM server(s)/client(s));
- Executor module: is responsible of deploying the required actions planned by the Planner module.

Finally, the Knowledge Base contains graph models of current and past infrastructure and DM service configurations.

B. ASM Work Methodology

The autonomic feature of the ASM is based on MAPE-K loop (in Fig. 2). It has three inputs: DM system/ Infrastructure models and the selected objective defined manually by the System administrator at initial step, named *Design time*. The infrastructure/DM graph models and the initial placement solution for DM system, along with the fore-coming graphs and placement solutions, will be stored in the ASM's Knowledge base. The output of the ASM is the DM servers that needs to be launched/stopped on particular infrastructure nodes along with their binding with their clients.

These graph models serve as an input to a CP solver that is integrated into the analyzer step, i.e., (A in Fig. 2). In this work, we consider the constraint satisfaction programming for solving the DM placement problem which provides at design time a feasible placement of the DM system within current infrastructure. However, this module is generic in that different placement approaches could be selected for both mapping/remapping steps.

Thereafter, the System evolution will be gathered by the Monitor which involves- based on the adopted scenario: (i) changes in the underlying infrastructure (e.g., node arrival/departure, resources loads), (ii) changes in the DM system (e.g., clients arrival/ departure), and (iii) other system administrator notifications, such as updating the solver objective to accommodate an urgent security campaign firmware updates instead of "normal" firmware/ configuration updates (M in Fig. 2).

The monitored information is passed to the analysis step, whenever a new change is observed. The analyser takes into account the last system state stored in its knowledge base. As a result, a new graph(s)- that reflects the new requirements and changes- will be generated and the constraint solver will be called to decide on the mapping of the new system's elements. Thus, the (Re)mapper module will be called first at the design step and then whenever a new information is observed and transmitted by the Monitor.

After receiving this information, the planner (P in Fig. 2)) produces a set of actions to be applied to move from the current system state to the target one by extracting the difference between the current and the target mapping, to do only necessary actions (e.g., decide the required number of server/client to be started/stopped). Finally, the Execution step applies these actions (E in Fig. 2)).

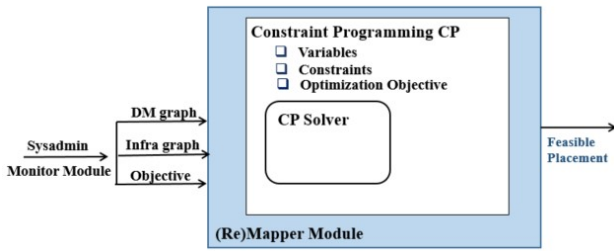


Fig. 4: Integrating CP in the ASM MAPE-K architecture

This whole process, after the design step, will be repeated in a loop based on the monitored information and the runtime conditions imposed by the Administrator. We further show in Fig. 4 how the constraint programming is integrated within ASM (Re)mapper.

Finally, in the case that the monitor detects a reconfiguration need and a CP solver is called, we may obtain that no feasible placement solution is found, for example, because there are no enough resources in the infrastructure while satisfying all required constraints. Thus, the Remapper module might need to wait for some time, so that other services release some infrastructure resources or some clients leave to try again with the solver. In such case, the ASM will keep the last solution and a notification will be sent to the system administrator.

C. Supported Use-Cases

The need for DM system reconfiguration is based on the targeted evolution type. Different changes may result in a need for a new system configuration in the DM context. Privacy is an important concern that is adopted in this work [11], where a client may need to be managed by a specific DM server or a DM server placed in a specific area due to some privacy considerations (e.g., the IoT device(s) belongs to a private enterprise or for a VIP (Very Important Person)). Thus, whenever new IoT devices ask to join the system and get registered and managed by a DM server, these newcomers will need some resources (e.g., CPU (Central Processing Unit) and RAM (Random Access Memory)) along with a specific privacy constraint. Further, IoT devices might leave the system so we need to stop the related DM clients, which results in reducing the pressure on the currently running servers.

Any DM system evolution at runtime observed by the Monitor (through an external notification issued by the administrator) will be reported to and analyzed by the Remapper. In case new devices arrival, ASM will individually check their privacy requirements to see if they can be registered with one of the existing servers. If the current servers cannot satisfy the new devices, a new server(s) need to be added the DM system, so a new DM graph will be generated to reflect the new system and the constraint solver will be called to decide on placement of the new graph.

While privacy is considered in our work to trigger the scaling process, but it is just a use-case, as there are other situations that might require adaptation and that can be handled by our proposed architecture. For example, adding more DM

servers to adapt to the increased number of IoT devices communicating with a single server, and then balancing the load among these servers instances represent a new motive for scaling to avoid the stress on one server and efficiently manage the huge device fleet. In such scenario, any information leading to redeployment/instantiation need will generate a new DM graph and launch again the CP solver.

The inform storm represents a real life use-case when a huge amount of IoT devices reboot at one time, e.g., after a power breakdown. Another use-case is the smart vehicles [26] which consists of numerous sensors and actuators that automate various tasks, such as traffic monitoring and braking. In this scenario, it is necessary to keep the vehicle's firmware up-to-date, and to provide a convenient solution to the problem of installing a new firmware that could be released to fix bugs or provide new functionality. A significant challenge with this use-case is the mobility, where a vehicle may disconnect from a certain DM server placed on a specific Edge node and require to reconnect to an alternate DM server according to its current geographic location. By automatically scaling new DM servers and placing them close to the moving vehicle, this mobility problem can be handled.

V. DM SERVERS PLACEMENT PROBLEM

The mathematical formulation of the placement problem given in [20] is strongly aligned with our DM service placement problem. Therefore, we have benefited from this model and modified the following parts to comply with our needs. In the first update, we have changed the definition of the proposed locality constraint (See Sec. V-B). Instead of enforcing some service components to be placed on specific physical nodes, we have introduced the dedicated zones to which each physical node belongs. Then, each DM component will have a privacy requirement regarding the accepted zone of the hosting node.

Another variation is the optimization objective. Here, we have proposed a multi-objective optimization to decide at runtime which objective we need to optimize (more details in Sec. V-C). Further, the CP solver is integrated in a feedback loop in order to make adaptation decisions at runtime.

A. System Model

1) *Infrastructure model*: It is defined as a directed graph $G^I = \langle H, \epsilon \rangle$, where H is the physical nodes (e.g., Edge and IoT devices) and $\epsilon = H \times H$ is the network links between the nodes : $\forall h \in H; \exists U(h)$ where $U(h) = (CPU, RAM, DZ)$

In the infrastructure graph, we defined for each node its available CPU and RAM resources in addition to its own dedicated zone. For the physical links, we define the available bandwidth and latency as follows:

$\forall e_{i,j} \in \epsilon; \exists LAT(e_{i,j})$ and $BW(e_{i,j})$, where $LAT(e_{i,i} = 0)$ and $BW(e_{i,i} = \infty)$ (In future work, the link type can be added as an additional characteristic).

2) *DM System Model*: We consider that our DM system represents as a service and could be modeled as a graph $G^a = \langle C, L \rangle$, where C is the components of our DM

system and $L = C \times C$ is the communication links between these components.

The nodes of this graph represent the DM servers and clients along with their resources requirement, namely CPU, RAM, and privacy :

$\forall c \in C; \exists u(c)$, where $u(c) = (Rcpu, Rram, Rdz)$ The $(Rcpu, Rram)$ are the resource requirements of each DM components and Rdz is the privacy requirement where each DM system's components need to be placed at some geographical place (e.g., customer premises for privacy issue).

In addition, for the links needed for the communication between the servers and the clients, we define the following requirements: $\forall k \in L; \exists Reqlat(k)$ and $Reqbw(k)$ which is the latency and the bandwidth requirements of the DM system communication links.

3) *Placement*: After modeling both the physical infrastructure and the DM systems as graphs, the next step is to decide where to place all DM system's components and the communication links on the infrastructure physical nodes and links, respectively.

The placement solution must satisfy all the resources requirements of the DM components (namely, CPU, RAM, BW and Latency) while, at the same time, respecting the underlying infrastructure available resources. Based on methods provided in the related work we decide to use the constrained programming approach for solving the placement problem (this choice was justified in the Sec. II).

B. Problem Formulation with Constraint Programming

Here we present the formulation model for the DM placement problem. Inspired by the related work [20], for the technical needs of the modeling (to ensure we have a full connected graph), the infrastructure graph is increased by adding a super-sink node (α), with unlimited resources (CPU, RAM...), to which all graph nodes can access. The links that connect the infrastructure nodes to α , and α to itself, have an infinite capacity. In the following, the variables related to both nodes and links are declared. $\forall k \in L : s = \{s_k \mid k \in [1, |L|]\}$ where s_k is the physical node that hosts the source component of a DM system link (k), where $(s_k \in H)$.

$t = \{t_k \mid k \in [1, |L|]\}$ where t_k is the physical node that hosts the target component of a link (k), where $(t_k \in H)$.

$n = \{n_{k,j} \mid k \in [1, |L|], j \in [1, |H|]\}$ where $(n_{k,j} \in H)$ is the physical node at position (j) in the path of link (k).

$h = \{h_i \mid i \in [1, |C|]\}$ where $(h_i \in H)$ is the physical node that hosts a component (i).

$p = \{p_k \mid k \in [1, |L|]\}$ where $(p_k \in H)$ is the position of the target component(t_k) in the path (n_k).

$a = \{a_{k,j} \mid k \in [1, |L|], j \in [1, |H|]\}$ represents a physical link between $(n_{k,j})$ and $(n_{k,j+1})$ in the path of a DM system's link (k), where $(a_{k,j} \in \epsilon)$.

$b = \{b_{k,j} \mid k \in [1, |L|], j \in [1, |H|]\}$ represents the bandwidth of the physical link ($a_{k,j}$).

Finally, l represents the physical link latency ($a_{k,j}$), where $l = \{l_{k,j} \mid k \in [1, |L|], j \in [1, |H|]\}$

After defining our problem model variables, we present the necessary constraints to be applied to control the possible combinations of values that these variables could obtain.

Constraint 1: *BINPACKING* constraints (knapsack-based reasoning [27]). Here we ensure that the sum of all mapped components demands does not exceed the maximum available CPU and RAM capacities of the infrastructure nodes.,

$$BINPACKING (\langle h, Reqcpu \rangle, CPU)$$

$$BINPACKING (\langle h, Reqram \rangle, RAM)$$

Constraint 2: To fix a DM system's component (i) to a specific location or DZ, we use this locality constraint:

$$DZ(h_i) = Loc(i), \forall i \in C$$

Constraint 3: Node at position (0) in the path (n_k) hosts the source component of a DM communication link (k).

$$n_{k,0} = s_k$$

Constraint 4: Node at position (p_k) in the path (n_k) hosts the target component of a DM communication link (k).

$$n_{k,p_k} = t_k$$

Constraint 5: When the source and the sink components are the same, they will be hosted on the same physical node.

$$s_k = t_k \leftrightarrow p_k = 1$$

Constraint 6: To avoid cycles in a path (n_k), the *ALLDIFFERENT* filtering algorithm [28] is used to prevent similar values for the variable ($n_{k,j}$).

$$ALLDIFFERENT(n_{k,j}, \forall j \in \{1, \dots, |H|\})$$

Constraint 7: Any path (n_k) ends with at least one occurrence of α (the super-sink added node to the infrastructure graph). For that, the *REGULAR* constraint [29] is added to ensure that the corresponding sequence of values taken by variables belong to a given regular language.

$$REGULAR(n_k, "[^\wedge\alpha] + [\alpha] + ")$$

Constraint 8: When two DM communication links (k, k') share a same component, then the physical node that hosts the target component of the first link will be the same node the hosts the source component of the second link. $t_k = s_{k'}$

Constraint 9: This constraint is a *BINPACKING* constraint to respect the bandwidth limit of each physical link.

$$BINPACKING (\langle b, Reqbw \rangle, BW)$$

Constraint 10: These last two constraints concern respecting the latency of each DM communication link, and the end-to-end latency along the path of the DM communication link, respectively:

$$l_{k,j} = LAT(a_{k,j}), \forall k \in L, \forall j \in \{0, \dots, |n_k|\}$$

$$\sum_{j=0}^{|n_k|} l_{k,j} \leq Reqlat(k), \forall k \in L$$

Moreover, in the DM context, the communication link type is an important requirement to be considered when solving the placement problem, where various communication technologies between devices are used such as Wi-Fi, Bluetooth, and ZigBee (based on the IoT device). Thus, when solving the placement problem, the link type could be added to our Infrastructure/DM models to better fit the DM context.

C. DM Optimization Function

Different objective functions have been introduced in the literature (see Sec. II for more details) of placement problem. However, the question that comes to mind is that: **What are**

TABLE I: Objective Optimization Classification

Metric	Description
Resources	Minimize the number of used Nodes
	Minimize the total used links' BW
Load balancing	Maximize each physical node minimum remaining CPU cores
	Maximize each physical link minimum remaining BW
Latency	Minimize the latency for urgent security patches

the optimization metrics to consider in the case of DM service provided to IoT users? To answer this, we provide a simple classification for placement objectives metrics that could fit for the DM context in the Table I. The optimal management of infrastructure resources is an important metric. With the objective *Resource*, we aim to minimize the computational or network resources used. For example, the DM system involves communications between servers and their bulk clients and communication traffic will pass through network links, that are shared with other services/applications, so minimizing network consumption is important for the DM context. However, this goal could imply that some nodes/links are overloaded while others are underloaded.

With the *Load balancing* objective, we aim to reduce the stress on both physical nodes and links (i.e., congestion rate) by balancing the load between the participant nodes and links in our infrastructure by choosing the least loaded nodes and links in the infrastructure. Finally, with the *Latency* objective, given that one of DM's features is to send security patch updates, it becomes necessary to ensure the quick arrival of such updates. To ensure this, we need to minimize the latency between the clients and their own servers.

From the discussion above and given the DM system features, there is no single objective function to be considered statically. Rather, it appears that this function depends on the chosen scenario and the actual communication information exchanged between the servers and their clients. To do this, a multi objective function is introduced with *coefficients* variables (α, β, \dots , etc.) that will be activated/deactivated at run-time by getting the value of 1 or 0, respectively. We formulate the final placement objective as follows:

$$Objective = \alpha Obj1 + \beta Obj2 + \dots$$

For our proof of concept and experimental validation we choose between these objectives: (i) Minimizing the total number of participating nodes, and (ii) Minimizing the total consumed bandwidth. Our optimization objectives are formulated as follow: $Obj1 = Min \sum_{i=0}^{|H|} h_i$, where ($h_i \in H$) is the physical node that hosts a component (i); $Obj2 = Min \sum_{j=0}^{|n_k|} b_{k,j}$, where $b_{k,j} = BW(a_{k,j})$, $\forall a_{i,j} \in \epsilon$.

VI. EXPERIMENTAL VALIDATION

We present here our experimental setup to assess our approach's ability to automatically adapt by horizontal scaling the DM servers number based on both the DM system and the infrastructure's evolution. First, we present our use case and

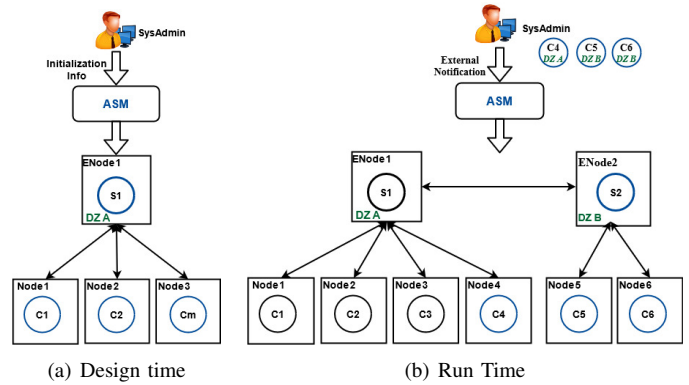


Fig. 5: Privacy Use Case, DM System Evolution

then detail the technical architecture of our setup. Then, we present our environment and ASM implementation.

A. Target Use Case

To assess our architecture, we considered the challenge of preserving the privacy of users who use a set of smart objects connected to each other and, potentially, to other users' objects. In such a case, IoT objects can use and propagate a lot of information about the features they offer and provide sensitive information about the users that they do not intend to reveal, e.g., knowing the features of most adopted objects by users. In this case, users may require their devices to be managed by an authorised server or by a server located in an authorised geographical place.

Therefore, we consider the following use case. Initially (See Fig. 5(a)), the DM system administrator will start a DM server in a specific location in the infrastructure and let the DM clients register to it as it meets their privacy requirements (i.e., based on the server locality). From here, the ASM will be in charge of managing the DM system throughout its life cycle.

After some time, new devices will arrive and ask to join the fleet and register with one of the servers respecting their privacy constraints. The clients information and requirements will be sent to ASM by the administrator via external notifications (See Fig. 5(b)). ASM will analyze this new information and then decide on the right actions to move from the system's current state to a new one that meets all DM components requirements. This involves either connecting each new client to an existing server that meets the privacy constraint, or else by instantiating a new server on a node within an acceptable location, i.e., dedicated zone.

B. Experimental Setup

The ASM inputs are the infrastructure and the DM system models which are generated based on the use case introduced in Sec.VI-A. This subsection details their attributes.

a) *Physical Infrastructure*: The global infrastructure is composed of: (i) 20 Edge nodes with CPU cores between 2 and 12 and available RAM between 2 and 24 GB, (ii) 40 less powerful extreme edge nodes close to IoT devices, namely customer premises nodes, characterized by CPU cores

between 1 and 2 and available RAM between 1 and 2 GB, and (iii) up to 1000 links that connect these nodes randomly with bandwidth up to 5/20 Gbps and latency up to 1/5 ms for the physical links that connect the edges nodes and 1/5 Gbps and latency up to 10/20 ms for the physical links that connect edges nodes to customer premises nodes where each node belongs to a specific zone (for the need of privacy scenario).

The nodes zones is given randomly where $DZ1$ to $DZ6$ is reserved of Edge nodes that will host DM servers and $DZ10$ to $DZ20$ is reserved of customer premises nodes.¹

b) DM Service: For the initial setup we start the DM service with one server and 3 clients managed by this server. After that, the administrator external notifications concerning new IoT clients will arrive following an exponential distribution of mean T_{IA} , where T_{IA} is the mean inter-arrival time of clients arrival notifications (measured in arbitrary time unit). Each DM client has a service time, i.e., the time the client remains in the system is randomly selected following an exponential distribution of mean S . Any notification that cannot be satisfied directly will be kept in a queue for some predefined time, namely Time To Live (TTL), waiting for some resources to be released and become available.

In total, our synthetic external notifications workload for the simulations contains request arrivals made of 40/60/80/100 arbitrary clients. The number of clients that arrive or leave is selected randomly between 1 and 4 client at each notification.

The communication between servers and their clients is done via bindings. The processing, RAM, latency and bandwidth requirements for DM components and binding is chosen in random manner with respect to the available infrastructure resources. Moreover, each DM component requires to be started on a specific dedicated zone within the physical infrastructure (more details in Sec. V).

C. Architecture Implementation

We detail here our choices in terms of architecture and language for the development, illustrated in Fig. 6. The software and modules run locally on a workstation with four physical cores and eight logical computing units of the Intel x86-64 architecture, supported by 32GB of DDR4 with 2133MHz RAM with Linux Debian 10 operating system.

For the mapper module, which solves the DM placement problem based on CP, we have chosen Choco as a solver inspired by the works [20] [16], which is a free and open-source Java library dedicated to constraint programming. The placement problem must be modeled in a declarative way by defining the set of constraints to be satisfied in each solution. Then, it will be solved by alternating constraint filtering algorithms with a search mechanism.

The three remaining modules of the ASM, are developed from scratch in Python. The choice of Python is motivated by its wide collection of libraries and its native compatibility with

¹The processing and the memory capacities for each infrastructure node are chosen randomly from set of values inspired by the most common OpenStack and Amazon EC2 instance types and the links bandwidth and latency values also and number of dedicated zones.

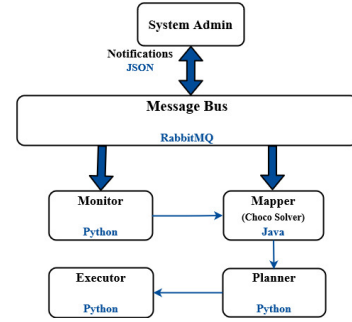


Fig. 6: The Technical Implementation of the Architecture

the JSON [30] data format used by the other components of our architecture (messaging bus). The ASM modules communicate with the system administrator via a common RabbitMQ messaging bus [31]. The RabbitMQ bus allows communication via Message Queuing Telemetry Transport (MQTT) which is a messaging protocol widely used in the context of IoT and compliant with Python.

D. Experimental Results

Our defined simulation scenario aims at initiating a DM service on a physical infrastructure. Later on, at runtime, external notifications about fleet evolution will be launched by the system, triggering our automatic manager to take scaling decisions of the DM servers. All the following experiments were repeated 5 times using 5 different notification workloads with the same parameters.

1) ASM Adaptation Characteristic:

We evaluate here how the ASM can adapt the number of DM servers at runtime to fulfill the new DM clients privacy requirements with a total number of 80 clients that want to join the DM system. Figure 7(a) shows the evolution of number of participated servers to adapt to evolution of DM clients through the time. It starts with the initial configuration of 1 server and 3 clients. From this figure we can see an increase in number of servers as new clients join the system (based on their privacy constraint on the acceptable server). However, when new clients arrive and their privacy requirements can be satisfied by the already running servers there is no need to scale out the DM servers. This explains the situations in the figure where we have new incomers and the number of servers does not increase. For example at the time point of 14 to 15 min, the number of servers is 4 while the number of clients in the system goes from 9 to 13 clients.

On the other hand, when the clients number decreased as they leave the system, Fig. 7(a) shows decreasing in the number of servers. This happens when a server has no more clients, e.g., at the time point of 1 min the ASM scale in the number of servers goes from 3 to 2 servers as the the number of clients goes from 9 to 6 clients. However, not all clients departure leads to ASM taking scale in decisions, as their server still have other clients to manage.

2) Acceptance Ratio:

We study the impact of total number of clients that arrived to the system on the ability to satisfy their joining requests

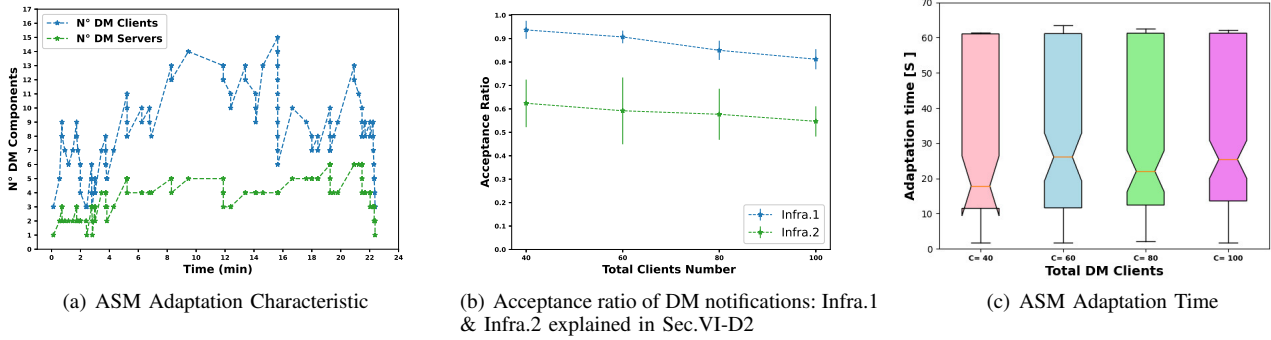


Fig. 7: Experimental Results

and placing them into the Infrastructure. To that aim, we use the acceptance ratio defined as the number of accepted clients arrival notifications over the total number of arrival notifications. Fig. 7(b) shows the acceptance ratio with respect to 4 different total number of clients that ask to join the DM system with two different infrastructures: (i) Infra.1: composed of 20 Edge nodes & 80 customers premises nodes, and (ii) Infra.2 with 20 Edge nodes & 40 customers premises nodes.

In general, and for both infrastructures, we can expect the acceptance ratio to decrease as the total clients number increases since more clients require devoting more physical resources to place the new clients and server when needed. This trend is confirmed by Fig.7(b). However, the decreasing is negligible and this can be explained by the fact of using a waiting queue for the unaccepted notifications to try again when other clients leave the system and some resources become available. Thus, using this queue allows increasing the acceptance ratio and preventing a sudden drop in its value when the number of clients increases from 40 up to 100 clients. However, with more powerful underlying infrastructure (the blue curve in Fig.7(b)) leads to more acceptance ratio as the network can handle more clients before becoming overloaded. It is worth mentioning that even with 40 clients, we do not have an acceptance rate of 1 as the combination of many constraints, namely CPU/RAM/Bandwidth and privacy, prevents the solver from finding a valid solution.

3) ASM Adaptation Time:

To be acceptable, the time spent by ASM to adapt to DM runtime evolution must be at most of the same order of magnitude as the deployment of the VMs that will host the DM components to not impact the deployment time of the DM service. This time includes the generation of a new DM configuration file, taking scale In/Out decisions and finding an acceptable placement of DM components. For that reason, we force the Choco solver in the Mapper to try to find an optimal solution in no more than 1 minute. It is in the same order of magnitude of the typical time to deploy and boot virtual functions in data centers [32].

Figure 7(c) shows the whisker plot of ASM adaptation time to the system administrator external notifications times for 4 different number of DM clients. It shows that the needed time

by the ASM to adapt to runtime notifications increases rather linearly with number of DM clients and never exceeds one minute, which is affected by the Mapper module time to find the placement solution. This rather linear increase is because an increase of DM clients number incurs a proportional increase in the total number of DM components. These clients do not come at once, but rather in a random group from 1 to 4 clients together. Thus, the size of the placement problem as the size of the DM components is not impacted.

The spread between median and lower quartile is smaller than the spread between median and upper one as most of placements require more time to find a solution of the new DM configuration or to reject the notification and send it to the waiting queue.

Knowing that the number of arrived clients significantly affects the size of problem, and therefore the needed time by solver to find a solution, this case should be considered as a scalability limitation to our solver. The constraint programming solver could not be fast enough and a new heuristics might be considered to find a near optimal placement.

To best of our knowledge this is the first work on distributed IoT DM at the Edge of the network. Thus, many parameters are set with random values and affect our simulation results. The first parameter is the *TTL* value: more *TTL* could increase the acquired acceptance ratio as more resources will be released and become available, but more time is needed to complete a full workload (long waiting queue). The solver maximum allowed time has also a direct impact on both acceptance ratio and ASM adaptation time. Giving more time to Choco solver may increase the acceptance ratio. However, since we are targeting client requests at runtime, it is very important to make a decision as quickly as possible. Moreover, giving more time to the solver cannot guarantee that a solution will be found as the infrastructure might be already overloaded.

Furthermore, since all workloads are randomly generated due to the unavailability of real-world workload- i.e. random arrival rate, service time and resources requirements - running more workloads lead to more consistent results and avoid some ambiguous results, e.g., the median value of the whisker plot for the clients number equal to 60 in Fig. 7(c).

VII. CONCLUSIONS AND PERSPECTIVES

We have addressed the main IoT DM challenges, namely Heterogeneity, dynamicity and scalability. For that purpose, we proposed a self-adaptive DM architecture for IoT with an autonomic manager that is capable of self-scaling the number of DM servers to the fleet changes and requirements at runtime through the distribution of DM operations at the Edge. This autonomic manager relies on a constraint programming solver that is integrated in a feedback loop to decide on DM servers and clients placement while optimizing the infrastructure resources usage. Further, we evaluated this architecture through simulation to the fleet evolution at runtime with a privacy as the target scenario. The results show that our manager is fast enough- only 1 minute- such that one can consider using it in a real environment to handle IoT fleet composition changes at run time and without the need of prior knowledge on the new IoT devices requirements.

For future work, we are investigating our proposal's scaling capability with respect to another scenarios such as physical resources thresholds and DM servers limitations. Also, from the point of view of constraints and models, we want to consider more dynamics (e.g. speed and acceleration of variations) in relation with Control Theory [33]. For interoperability motive, a constraint regarding the management protocol used by the DM server could be considered. Another important enhancement will be testing this architecture with real infrastructure e.g., Grid5000 [34] FIT IoT-LAB Testbed [35].

REFERENCES

- [1] F. Aïssaoui, S. Berlemont, M. Douet, and E. Mezghani, "A semantic model toward smart iot device management," in *Workshops of the International Conf on Advanced Information Networking and Applications*, (Caserta, Italy), pp. 640–650, Springer Publishing, 2020.
- [2] T. Perumal, S. K. Datta, and C. Bonnet, "Iot device management framework for smart home scenarios," in *2015 IEEE 4th Global Conf on Consumer Electronics (GCCE)*, (Japan), pp. 54–55, IEEE, 2015.
- [3] K. Shea, "Device management in the internet of things—why it matters and how to achieve it," <http://www.new-techeurope.com/2017/06/07/device-management-internet-things-matters-achieve/>, 2017. Accessed on 2021-02-20.
- [4] A. Computing *et al.*, "An architectural blueprint for autonomic computing," *IBM White Paper*, vol. 31, no. 2006, pp. 1–6, 2006.
- [5] F. Rossi, P. Van Beek, and T. Walsh, *Handbook of constraint programming*. USA: Elsevier, 2006.
- [6] B. Forum, "Tr-069 cpe wan management protocol." https://www.broadband-forum.org/download/TR-069_Amendment-6.pdf, 2018. Accessed on 2021-02-23.
- [7] O. M. Alliance, "Lightweight machine to machine technical specification," *Approved Version*, vol. 1, no. 1, 2017.
- [8] M. Elkhodr, S. Shahrestani, and H. Cheung, "The internet of things: new interoperability, management and security challenges," *arXiv preprint arXiv:1604.04824*, vol. abs/1604.04824, p. 85–102, 2016.
- [9] J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang, and W. Zhao, "A survey on internet of things: Architecture, enabling technologies, security and privacy, and applications," *IEEE Internet of Things Journal*, 2017.
- [10] Z. Wen, R. Yang, P. Garraghan, T. Lin, J. Xu, and M. Rovatsos, "Fog orchestration for internet of things services," *IEEE Internet Computing*, vol. 21, no. 2, pp. 16–24, 2017.
- [11] J. H. Ziegeldorf, O. G. Morchon, and K. Wehrle, "Privacy in the internet of things: threats and challenges," *Security and Communication Networks*, vol. 7, no. 12, pp. 2728–2742, 2014.
- [12] N. Ayeb, E. Rutten, S. Bolle, T. Coupaye, and M. Douet, "Towards an autonomic and distributed device management for the internet of things," in *IEEE 4th International Workshops on Foundations and Applications of Self* Systems*, (Sweden), pp. 246–248, IEEE, 2019.
- [13] N. Ayeb, E. Rutten, S. Bolle, T. Coupaye, and M. Douet, "Coordinated autonomic loops for target identification, load and error-aware device management for the iot," in *15th Conference on Computer Science and Information Systems (FedCSIS)*, (Bulgaria), pp. 491–500, IEEE, 2020.
- [14] J. O. Kephart and D. M. Chess, "The vision of autonomic computing," *Computer*, vol. 36, no. 1, pp. 41–50, 2003.
- [15] M. Tahir, Q. M. Ashraf, and M. Dabbagh, "Towards enabling autonomic computing in iot ecosystem," in *IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress*, (Japan), IEEE, 2019.
- [16] F. A. Salaht, F. Desprez, and A. Lebre, "An overview of service placement problem in fog and edge computing," *ACM Surveys*, 2020.
- [17] B. Donassolo, *IoT Orchestration in the Fog.(L'orchestration des applications IoT dans le Fog)*. PhD thesis, Grenoble Alpes University, France, 2020.
- [18] S. Challita, F. Paraiso, and P. Merle, "A study of virtual machine placement optimization in data centers," in *7th International Conference on Cloud Computing and Services Science*, (Porto, Portugal), pp. 343–350, INSTICC, 2017.
- [19] B. Donassolo, I. Fajjari, A. Legrand, and P. Mertikopoulos, "Load aware provisioning of iot services on fog computing platform," in *ICC IEEE International Conf on Communications*, (China), pp. 1–7, IEEE, 2019.
- [20] F. A. Salaht, F. Desprez, A. Lebre, C. Prud'Homme, and M. Abderrahim, "Service placement in fog computing using constraint programming," in *IEEE International Conf on Services Computing*, (Italy), IEEE, 2019.
- [21] Y. Xia, *Combining Heuristics for Optimizing and Scaling the Placement of IoT Applications in the Fog*. PhD thesis, Université Grenoble Alpes, 2018.
- [22] C. Prud'homme, J.-G. Fages, and X. Lorca, "Choco solver documentation," *TASC, INRIA Rennes, LINA CNRS UMR*, vol. 6241, 2016.
- [23] L. Perron and V. Furnon, "Google's or-tools," 2019.
- [24] P. Laborie, J. Rogerie, P. Shaw, and P. Vilím, "Ibm ilog cp optimizer for scheduling," *Constraints*, vol. 23, no. 2, pp. 210–250, 2018.
- [25] S. R. Department, "Internet of things- active connections worldwide 2015-2025." <https://www.statista.com/statistics/1101442/iot-number-of-connected-devices-worldwide/>, Jan 2021. Accessed on 2021-02-23.
- [26] K. Fizza, N. Auluck, A. Azim, M. A. Maruf, and A. Singh, "Faster ota updates in smart vehicles using fog computing," in *Proceedings of the 12th IEEE/ACM International Conference on Utility and Cloud Computing Companion*, (New York, NY, USA), pp. 59–64, Association for Computing Machinery, 2019.
- [27] P. Shaw, "A constraint for bin packing," in *International conference on principles and practice of constraint programming*, (Berlin), pp. 648–662, Springer, 2004.
- [28] J.-C. Régim, "A filtering algorithm for constraints of difference in cpsp," in *AAAI*, (USA), pp. 362–367, American Association for AI, 1994.
- [29] G. Pesant, "A regular language membership constraint for finite sequences of variables," in *International Conf on principles and practice of constraint programming*, (Heidelberg), pp. 482–495, Springer, 2004.
- [30] F. Pezosa, J. L. Reutter, F. Suarez, M. Ugarte, and D. Vrgoč, "Foundations of json schema," in *Proceedings of the 25th International Conference on World Wide Web*, (Republic and Canton of Geneva, CHE), pp. 263–273, International World Wide Web Conferences Steering Committee, 2016.
- [31] A. RabbitMQ, "Messaging that just works - rabbitmq." 2020. Accessed 07-June-2021.
- [32] M. Mao and M. Humphrey, "A performance study on the vm startup time in the cloud," in *2012 IEEE Fifth International Conference on Cloud Computing*, (Honolulu, HI, USA), pp. 423–430, IEEE, 2012.
- [33] M. Litoiu, M. Shaw, G. Tamura, N. M. Villegas, H. Müller, H. Giese, R. Rouvoy, and E. Rutten, "What Can Control Theory Teach Us About Assurances in Self-Adaptive Software Systems?," in *Software Engineering for Self-Adaptive Systems 3: Assurances*, vol. 9640, Springer, May 2017.
- [34] D. Balouek, A. C. Amarie, G. Charrier, F. Desprez, E. Jeannot, E. Jeanvoine, A. Lèbre, D. Margery, N. Niclausse, and L. Nussbaum, "Adding virtualization capabilities to the grid'5000 testbed," in *International Conf on Cloud Computing and Services Science*, pp. 3–20, Springer, 2012.
- [35] C. Adjih, E. Baccelli, E. Fleury, G. Harter, N. Mitton, T. Noel, R. Pissard-Gibollet, F. Saint-Marcel, G. Schreiner, J. Vandaele, and T. Watteyne, "Fit iot-lab: A large scale open experimental iot testbed," in *IEEE 2nd World Forum on IoT*, (Italy), pp. 459–464, IEEE, 2015.

3rd International Forum on Cyber Security, Privacy and Trust

NOWADAYS, information security works as a backbone for protecting both user data and electronic transactions. Protecting communications and data infrastructures of an increasingly inter-connected world have become vital nowadays. Security has emerged as an important scientific discipline whose many multifaceted complexities deserve the attention and synergy of computer science, engineering, and information systems communities. Information security has some well-founded technical research directions which encompass access level (user authentication and authorization), protocol security, software security, and data cryptography. Moreover, some other emerging topics related to organizational security aspects have appeared beyond the long-standing research directions.

The International Forum of Cyber Security, Privacy, and Trust (NEMESIS'22) as a successor of International Conference on Cyber Security, Privacy, and Trust (INSERT'19) focuses on the diversity of the cyber information security developments and deployments in order to highlight the most recent challenges and report the most recent researches. The session is an umbrella for all cyber security technical aspects, user privacy techniques, and trust. In addition, it goes beyond the technicalities and covers some emerging topics like social and organizational security research directions. NEMESIS'22 serves as a forum of presentation of theoretical, applied research papers, case studies, implementation experiences as well as work-in-progress results in cyber security. NEMESIS'22 is intended to attract researchers and practitioners from academia and industry and provides an international discussion forum in order to share their experiences and their ideas concerning emerging aspects in information security met in different application domains. This opens doors for highlighting unknown research directions and tackling modern research challenges. The objectives of the NEMESIS'22 can be summarized as follows:

- To review and conclude research findings in cyber security and other security domains, focused on the protection of different kinds of assets and processes, and to identify approaches that may be useful in the application domains of information security.
- To find synergy between different approaches, allowing elaborating integrated security solutions, e.g. integrate different risk-based management systems.

- To exchange security-related knowledge and experience between experts to improve existing methods and tools and adopt them to new application areas

TOPICS

- Biometric technologies
- Cryptography and cryptanalysis
- Critical infrastructure protection
- Security of wireless sensor networks
- Hardware-oriented information security
- Organization- related information security
- Social engineering and human aspects in cyber security
- Individuals identification and privacy protection methods
- Pedagogical approaches for information security education
- Information security and business continuity management
- Tools supporting security management and development
- Decision support systems for information security
- Trust in emerging technologies and applications
- Digital right management and data protection
- Threats and countermeasures for cybercrimes
- Ethical challenges in user privacy and trust
- Cyber and physical security infrastructures
- Risk assessment and management
- Steganography and watermarking
- Digital forensics and crime science
- Security knowledge management
- Security of cyber-physical systems
- Privacy enhancing technologies
- Trust and reputation models
- Misuse and intrusion detection
- Data hide and watermarking
- Cloud and big data security
- Computer network security
- Assurance methods
- Security statistics

TECHNICAL SESSION CHAIRS

- **Awad, Ali Ismail**, Luleå University of Technology, Sweden
- **Bialas, Andrzej**, Research Network Lukasiewicz – Institute of Innovative Technologies EMAG, Poland

Heuristic Risk Treatment for ISO/SAE 21434 Development Projects

Christine Jakobs, Matthias Werner
TU Chemnitz
Chemnitz, Germany
christine.jakobs@informatik.tu-chemnitz.de
matthias.werner@informatik.tu-chemnitz.de

Karsten Schmidt, Gerhard Hansch
AUDI AG
Ingolstadt, Germany
karsten.schmidt@audi.de
gerhard.hansch@audi.de

Abstract—Due to new technologies for connectivity, automotive systems shift from a closed to an open system approach. Therefore, automotive systems have a rising demand for security, letting security be an upcoming field in research and practice. Also, the newly published process standard ISO/SAE 21434 demands adjustments in the development process to address cybersecurity. The unique characteristics of automotive systems leave many approaches from other system types inapplicable. This work concentrates on the risk treatment step in the cybersecurity development process. Due to the vast amount of differing terminology, we see the need to define a flexible taxonomy adaptable to several system types and used in systems with normative references. We use this taxonomy to develop a heuristic approach for risk treatment based on a distinct terminology for security requirements. The presented method is extendable to include several trade-off points.

I. INTRODUCTION

With rising interest in connectivity and Car2X, also automotive security gets into focus. ISO/SAE 21434 [1], the process standard for automotive security, defines the security analysis and risk treatment process into three steps. The process starts with a security relevance evaluation [2] for deriving the first criticality and filtering the targets of evaluation to relevant ones. After that, the risk analysis, followed by the risk treatment step, occurs.

Risk analysis is done on the decomposed system and tells a story about five questions. The question about what can go wrong determines possible damage scenarios, weighted according to their damage potential - how bad is this? The threat identification and attack path analysis show how damage scenarios happen. A defined attacker model answers the question about who can do that, which allows deriving the attack probability. The combination of both impacts (damage potential and required attack potential) determines the risk value and enables a criticality ranking.

Risk treatment uses the impacts and risk numbers to consolidate the story of the different risk analyses in the system. In this step, weighting the impacts allows a fine-granular prioritization of the risks. Defense method assignment leverages the impacts and thereby reduces the risks.

The first arising problem when looking for demands and approaches for risk treatment is the vast amount of different terminology. Unclear terminology makes it difficult to

understand demands and compare approaches. An example is UNECE No. R155 [3]. While its main section uses the term mitigation, the Annex uses security control, measure, and mitigation without proper definition. The same applies to other sources in the literature. Therefore, we see a need to find a distinct definition of the terms used for the risk treatment in automotive systems.

On the other hand, efficient risk treatment demands a structured data basis of the used mitigations. We ground our method on a simple taxonomy based on a distinct terminology with the possibility of transferring it to other system types. The developed defense method catalog is a cross-product of the taxonomy whereby the taxonomy is independent of concrete system information. The presented defense method catalog covers such system-type-related information.

Literature frequently covers risk treatment theoretically, but an efficient and flexible method is missing. Especially methods transferable to different input sources are rare. Most approaches are for particular systems, e.g., service-oriented and web-based systems. Others are implemented in a framework, demanding unique kinds of data sources. We aim for a flexible heuristic approach for risk treatment. Our approach uses general information from risk analysis rather than detailed system models, which makes the approach adaptable to practical settings.

Security requirement demands from system external sources, like normative or organizational policies, are often mixed with requirements related to specifics of the target of evaluation. After risk treatment, applied trade-offs must adhere to a system's external security demands. Otherwise, the system might be cost-effective but infeasible from a legal point of view. Therefore, we observe the origin of the requirements to enable traceability of the security demands.

Risk analysis determines threats to the system based on attack paths. Those have different configurations. Single threats (one-element attack paths) can be compared to single-points-of-failure in reliability or are preparation attacks for other paths. Prioritizing them in the treatment process first mitigates crucial threats (SPOF) and preparation attacks. The latter directly cut attack paths, reducing the effort of treating those. Updating the risks by determining the impact of the defense method on other risks raises the by-catch and reduces the

overall effort of implementing defense methods.

Risk treatment aims to minimize the risk of system threats by either reducing the impact or raising the required attack potential (reducing the attack probability). It is reasonable to prioritize the impacts in the treatment procedure for the efficiency of the process. Those priorities allow trade-offs between different impact categories or attacker model categories in a cost-pressure situation. Our approach aims for a flexible prioritization of the risk treatment procedure and the risk impact categories to allow different trade-off points.

Based on the relation to the automotive industry, the presented work starts with a short introduction of the normative references for risk treatment in automotive development (Section II). In Section III we introduce our taxonomy for security requirements. Section IV covers the approach for risk treatment. Please note that NDA reasons prevent an exhaustive evaluation of our approach. We tried to include several examples in the text, where appropriate. Section V discusses the limitations and possible further trade-offs for our risk treatment method. In Section VI we conclude our contribution and give an outlook for future work.

II. NORMATIVE REFERENCE

Two normative references are relevant for type approval in automotive systems: UNECE No. R155 [3] and ISO/SAE 21434 [1]. The demands of both norms have a different layer of detail.

A. UNECE No. R155

According to UNECE No. R155 [3] the OEM has to protect the vehicle type and to implement “all mitigations [...] which are relevant for the risks identified.” [3].

Besides the general demand for mitigation implementation, UNECE No. R155 has three concrete demands:

- Intrusion-Detection for the vehicles of a certain type
- A central monitoring facility for new threats and vulnerabilities
- The use of up-to-date cryptographic modules

Annex 5 provides a list of possible risks and appropriate mitigations. The mitigations listed in the Annex are not concrete methods, but rather categories of mitigations, e.g., “The vehicle shall verify the authenticity and integrity of messages it receives” [3]. The OEM may differ from the provided list of mitigations if it is insufficient to mitigate a certain risk.

In conclusion, UNECE No. R155 demands risk mitigations according to the provided categories in the Annex. The general demands are only OEM-related, one regarding the monitoring facility, and two vehicle-related demands. There is no suggestion regarding threat mitigation techniques or methods.

B. ISO/SAE 21434

ISO/SAE 21434 [1] describes the clauses in a triple of input, the requirements and recommendations, and the output. Input is the necessary and optional predecessor work products from other clauses. The requirements define the demands of the ISO

and provide possible methods or procedures. Output defines the work products resulting from this clause.

For the risk treatment, ISO/SAE 21434 demands to use the item definition (model of the target of evaluation), the identified attack paths, and the risk values as results from the risk analysis. Optional inputs are cybersecurity specifications (from former development or higher abstraction levels), previous risk treatment decisions, damage scenarios with their impact rating, and attack paths with feasibility ratings.

ISO/SAE 21434 requires the risk treatment for all identified risks by using one or more treatment options. Those options are the classical ones: risk avoidance and reduction, risk-sharing or retaining. The process documentation must record the decision to retain or share risk. Therefore, ISO/SAE 21434 explicitly allows risk acceptance up to a certain level, as long as this threshold and the retained risks are documented.

Like UNECE No. R155, the ISO does not provide possible methods for risk treatment.

III. SECURITY DEFENSE REQUIREMENTS

Our taxonomy of security requirements (see Figure 1) has three different categories: the origin, the type as well as the defense methods. The objective of the taxonomy of security requirements is the use in the risk treatment step. Therefore, the scope is on those requirements that apply to the vehicle and directly influence its behavior. Process requirements (e.g., audit, testing) and supporting processes (e.g., documentation) are related to the ecosystem in the development process and therefore excluded.

A. Related Work

In [4] the authors define several levels for the selection of defense methods. Their categorization is more detailed than our approach, e.g., to the circuit level. Since, in practice, the risk analysis for components is done based on a selected hardware platform, certain levels for hardware cannot be taken into account anymore. Therefore, we decided on more abstract and less detailed control categories. The basis for their approach to risk treatment is a rich and detailed data model. The data model allows a very comprehensive analysis of dependencies between defense methods. On the other hand, this also demands to use their entire framework for the security process. Otherwise, the demanded input data model may be impossible to acquire.

Pfleeger [5] describes a categorization of defense methods. The main categories are encryption, software, hardware, and physical. Typically, encryption is no stand-alone method but a feature of another defense method, e.g., secure boot, TLS. Physical controls like locks on doors are not applicable for vehicle security. Pfleeger integrates defense methods implemented as separate functions like intrusion detection systems and password checkers into software and hardware levels. We decided to differentiate between the level the control has its impact and the technical and functional control categories. Those include all applicable controls Pfleeger mentions, just in a different categorization.

Also [6] describes the categorization of security requirements into different layers. Those are regarding the point where the information lies: internally, externally, or both. Additionally, Chung differentiates between the development stage of the scope of the methods. We concentrate only on the run-time stage of the system and leave process demands aside. Procedural methods cannot be assigned to the system in the risk treatment step but accompany the complete development process. Therefore, they are in the scope of development process design, which is a separate topic.

The literature review of Akhunzada et al. [7] results in a thematic taxonomy regarding the different layers of software-defined networks. The categorization is very comprehensive but not transferable to other system domains. We aim for a more pragmatic taxonomy easily adaptable to other system types.

[8] presents another taxonomy based on literature research. The scope of this work is restricted to software integrity protection techniques. Different system views build the basis for the categorization: system view, defense view, and attack view. The evaluated literature is mapped onto the views and correlations are evaluated. Nevertheless, besides the different scopes, the granularity of the taxonomy is not helpful for our approach. We aim for a flexible approach to clarify the terminology needed for risk treatment to use it for the defense method assignment.

B. Origin

Security Requirements arise from different sources. Those are general conditions and system-related requirements. The former depict system external sources while the latter arises from the security analysis of the target of evaluation.

a) *General conditions*: General conditions are related to the vehicle type. They may not directly support accomplishing a security goal, but non-fulfillment provoke a mission failure.

The most critical source for those requirements is regulatory organizations. Norms like UNECE No. R155 [3] and ISO/SAE 21434 [1], but also country-specific regulations like GB/T 40856-2021 [9] demand certain types of security requirements in a varying level of detail. Examples are the intrusion detection system demand of UNECE No. R155. Those regulatory requirements are relevant for the type approval of the vehicle type. Therefore, a non-accomplishment endangers the possibility of selling the vehicle type, at least in certain countries.

Security-in-depth and security-by-design demand to define certain basic security standards on organizational level [10]. Those are also a source for general conditions. A non-accomplishment of the company policies does not lead to the loss of type approval but endangers the company's internal vehicle audit, e.g., during testing. Also, company policies may have a varying level of detail from distinct methods in a certain variation to general demands.

The nature of general conditions is that they have varying levels of detail. They may not explicitly project to specific defense methods but categories of those. Therefore, they may not directly support accomplishing a security goal (e.g.,

Confidentiality) to a certain degree but demand a defense method that supports this goal. One idea is to formalize them in a logic-based language to define the projections from the policies to their varying level of detail in the defense method catalog.

b) *System-related Requirements*: Requirements that arise from the target of evaluation are system-related. Those depict the necessity for defense methods against the risks evaluated in the risk analysis.

The system-related requirements are distinct since they base on specific risks according to threats against security goals and impacts regarding their damage and attack potential. Therefore, system-related requirements are refinable until they demand a defined variation of specific defense methods.

C. Types of Security Requirements

The type of security requirements defines the nature of the security requirements. The taxonomy categorizes them into those directly recognizable in the resulting system or not.

a) *Measures*: Measures are those requirements that are not directly recognizable in the system. They instead depict system design methods that can be non-technical, procedural, or logical methods against violating a security goal. Examples are the prevention of specific information flows or removing unused software libraries instead of applying expensive defense methods. Measures are typically general conditions or system-related requirements applied to the system during the development process. They are not directly used in the risk treatment step but are rules to verify after risk treatment.

b) *Controls*: Controls are requirements that are directly or indirectly recognizable in the system. Risk treatment refines those requirements till they depict specific defense methods in certain variations. Controls accomplish the means of risk treatment: reduce, detect, or avoid risks entirely. Examples are safeguards on all system levels like access control, Intrusion Detection Systems or secure communication protocols.

The different layers of the controls represent the defense-in-depth onion model: structural, technical, and functional controls. Structural controls are related to the overall system structure, e.g., network segmentation. They answer the question - How does the system structure look to avoid specific threats? Technical controls defend threats from the internal processing or component side. Examples are controls preventing the manipulation of software at rest or during run-time. Functional controls complete the onion model by providing defense on function level. Those controls define the behavior of connections to and within the system, e.g., with communication protocols and access controls.

D. Method categories

The last part of the security requirement taxonomy are the categories of defense methods. Those categories are dependent on the development project and are adjustable for every vehicle type in automotive development projects. This adjustment ensures that the categories are up-to-date and complete.

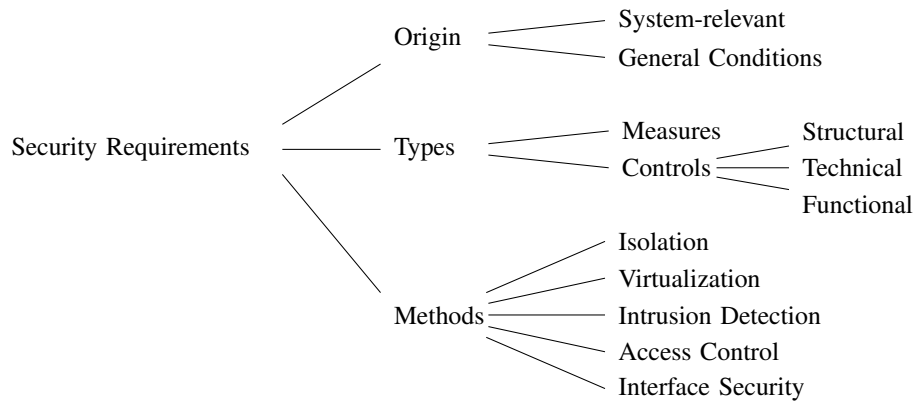


Fig. 1. Taxonomy of security requirements split into origin, types and methods.

In principle, the categories depict classes of defense methods and allow a more straightforward assignment in the risk treatment process. One suggestion is to align the classes with the general conditions, e.g., intrusion detection, isolation, and segmentation. Examples of typical classes for automotive development projects show Figure 1 without a claim for completeness.

E. Catalog of Defense Methods

The presented taxonomy allows a cross-product of types and method categories by assigning concrete defense methods (see Table I).

The assignment clusters the set of available defense methods into the categories from the taxonomy and annotates them with the type of control. Demands from general conditions like state-of-the-art cryptography do not lead to measures assigned to the list of defense methods. Those demands instead limit the scope of the available methods, e.g., non-appropriate cryptographic procedures are not part of the assignment.

Defense methods may be configurable, e.g., secure service-oriented communication (sSOA) with SOME/IP or SOCKS. Those variations can be directly included in the catalog or hidden as variants in the properties of the method. Encryption types are not direct defense methods but used by defense methods, e.g., secure communication protocols. They are therefore variants of the using method.

Defense methods may have no direct impact on the security goals. The impact may be emergent only in combination with other controls, e.g., a hardware security module (HSM) is only helpful in combination with a defense method that uses the cryptographic algorithms and the secure key storage provided by the HSM. Therefore, those methods are not included in the catalog but are variants of the defense method. An example would be TLS in combination with a present HSM [11]. In this case, there would be at least two variants of TLS, with and without an HSM present.

The catalog of defense methods is also subject to change. The catalog needs to be updated and refined in every development cycle, e.g., every vehicle type. Arising new threats

TABLE I
CLASSES AND IMPLEMENTATIONS OF DEFENSE METHODS. COLUMNS ARE THE TYPES OF CONTROLS (S=STRUCTURAL, T=TECHNICAL, F=FUNCTIONAL). ROWS INDICATE THE CLASS HIERARCHY AND EXAMPLES FOR CONCRETE IMPLEMENTATIONS. X INDICATES THAT THIS IMPLEMENTATION CAN BE USED FOR THIS CONTROL TYPE.

Implementations		Control Types		
		S	T	F
Isolation	NW Segmentation			
	VLAN	x		x
	Physical	x		
	Firewall	x	x	x
	Host Segmentation			
	CPU		x	x
Virtualization	SOA Domain			
	Hypervisor			
	Sandboxing		x	x
Intrusion Detection	Runtime			
	IDS	x	x	x
	Logging			x
	Runtime Protection		x	x
	Startup			
	Secure Boot		x	x
Authenticated Boot		x	x	
Interface Security	Communication			
	Secure SOA			x
	TLS			x
	IPsec			x
	SOK			x

may lead to changes during a development cycle. In this case, the responsible security engineers need to be informed about deleted methods and possible substitutions. This accounts also to other system types [10].

F. Properties of Defense Method

Defense methods mitigate threats to security goals. It is possible to derive the threat and damage scenario type from the targeted security goals. Therefore, it is better to assign the security goals as properties of the defense method rather

than the threat and damage scenario types. Following that, the impact on the different security goals is part of the properties. In [5] a range from -2 to +2 for each security goal is suggested. The positive part depicts the advantages of the defense method on security goals, while the negative part illustrates that defense methods may negatively influence certain security goals.

The definition of the effect on the attack potential and types of damage scenarios illustrates a defense method's influence on specific threats. This step reuses the method from the risk analysis, for the attack's potential impact may include the influence on the needed time, knowledge, tools, expertise, and access of the attacker.

The surface, as well as the dependency properties, limit the applicability of defense methods. The surface is regarding the attack surface, which this defense method mitigates. Dependencies relate to supporting processes needed from the technical side for this method (variant) to work or the limited scope of a defense method, e.g., VLAN technology is only applicable for Ethernet LAN.

It is possible to also assign costs for the defense method. Those represent the resource usage in the case of implementing this method. Examples would be processor clocks per byte in the case of DES [12]. In [11] different ways to derive the costs for cryptographic algorithms and examples are provided. Cost definitions allow to include risk treatment directly into resource scheduling procedures [12] which is out of scope in this work.

The different variants of a defense method have their properties assigned. Table II provides an example for securing JTAG. The two variants are disabling JTAG physically, which is the most secure variant but highly influences availability of the system. On the other hand is JTAG security by utilizing cryptography, e.g., SSL [13] which has a compared high effort.

TABLE II
EXAMPLE OF DEFENSE METHOD PROPERTIES FOR SECURING THE JTAG PORT WITHOUT CLAIM FOR COMPLETENESS. THE VALUES FOR COST, IMPACT AND EFFECT ARE VARIABLES TO ASSIGN BASED ON THE GIVEN SETUP.

	Property	Variants	
		Physical (Disable)	Cryptographic
	Technology	JTAG	JTAG
	Costs	0	?
Impact	Confidentiality	+2	+1
	Integrity	+2	+1
	Availability	-2	-1
	Surface	Local	
	Dependencies	-	SSL
	Effect	RAP: ?; DP: ?	

IV. RISK TREATMENT

The goal of risk treatment is twofold: Minimizing the risk, which means reducing the costs of non-implementation of defense methods, and minimizing the effort by reducing the costs of implementing defense methods.

In the security requirement taxonomy, general conditions are relevant for the type approval and/or necessary to fulfill OEM demands. Therefore, their costs for not-implementation are infinite. On the other hand, the costs for implementing general conditions are most of the time variable. They are typically related to categories of defense methods. Therefore, the implementation costs depend on the chosen method of the respective category. For system-related requirements, the costs of not-implementation need to be traded against the implementation costs. In the case of low-rated risks, it may be cheaper not to implement costly defense methods.

Also in settings where the direct implementation costs cannot be taken into account, risk treatment needs to take into account those cost-related problems. This can be done by reducing the overall number of defense methods through structured method assignment, e.g., by first cutting attack paths before treating the complete risk. Another possibility is to assign methods which impact several security goals at once instead of using a method per security goal.

The remaining section proceeds through the complete risk treatment process. Because of the already discussed problems with acquiring the costs of defense methods, we count for efficiency through a structured method assignment, and illustrate possible trade-offs and optimization points.

A. Prerequisites

According to ISO/SAE 21434 [1] risk treatment is related to identified risks and attack paths of the target of evaluation. Concrete, those risks whether the impact is against the road user. While it is possible to consider other impact categories, we will concentrate on those risks. The treatment of other risks is subject to by-catch of assigned methods and trade-offs.

The risks, threats, and attack paths, as well as their impact, are analyzed in the risk analysis step. There are different ways to complete this predecessor process step. Prominent are integrated approaches, e.g., MoRA [14] in the automotive world, or approaches developed for different sub-steps of risk analysis, e.g., attack trees [15] [16] [17] for attack path identification and rating.

For the presented approach, it is reasonable to evaluate the risk values in the risk analysis threefold: The risks against the road user, the OEM, and combined. Those three risks allow different options for trade-offs during or after method assignment. According to damage potentials and assumptions are outputs from the risk analysis steps in ISO/SAE 21434. Assumptions may be regarding the importance of specific damage scenarios and threats or related to other items' security process results. Therefore, they must be considered in the risk treatment because of their impact on threats and damage scenarios and traced for later verification steps.

B. Strategy

In general, risk treatment should be done holistically over the complete system. A global approach demands formal constraints for all defense methods and a distinct format for the input data. The latter is a big problem, especially in distributed

development environments like the automotive industry. Also, clearly defined tools for risk analysis allow different levels of abstraction and typically use natural language to formulate the damage scenarios and threats. Therefore, the bigger the input space for the risk treatment, the more significant the divergence between the input data. This divergence is also because the automotive environment's functionalities and components are diverse. The vehicle combines service-oriented functionalities with hard real-time classic development strands. This problem is minor on a lower level of abstraction, e.g., on the component level. Single components typically combine functionalities from similar development strands and complexity. Therefore, their analysis results are less diverse. Nevertheless, a global brute force approach would lead to a state-space explosion due to the nature of such NP problems.

Another strategy could be to assign all possible defense methods on component level to minimize the impact of a threat. On the one hand, less can be more depending on the defense methods. Assigning more defense methods leads to higher costs and might even lead to new risks to the system. Also, automotive systems are embedded systems with limited resources. Therefore, more defense methods than needed might leave the system infeasible.

An even lower level of abstraction would be assigning the defense methods in the decomposed system to each evaluation target individually. This strategy has the lowest probability of a state-space explosion and is accomplishable by brute force leading to an optimal solution for the different evaluation targets. On the other hand, a complete individual risk treatment might lead to various defense methods deployed to one component and its functionalities. The result is high costs for implementing defense methods and high resource usage, which might leave the system infeasible.

A good strategy for efficient and cost-effective risk treatment is a component-level heuristic approach. The defense methods are assigned where needed based on prioritizing the threats on the system. A by-catch test reveals a positive impact of the defense method on other component parts or functionalities.

a) Possible Heuristics: Related work reveals different possible heuristics for risk treatment. Ruddle et al. [15] present the straightforward idea of highest risk first. Risk is a combination of the damage potential and the required attack potential of the attack path and allows a global ranking of the relative priority. What is not possible is detailed heuristics which, e.g., prefer certain types of damage potentials.

In [10] a prioritization according to the highest impact or the highest likelihood first is suggested. Such a prioritization enables a ranking of risks according to the impact regarding their damage or the attack potential. Depending on the level of detail of the input model, this approach allows even heuristics with several layers, e.g., highest safety impact first.

A prominent approach for risk treatment are the defense-in-depth layers [10]. The idea is to assign defense methods from different types onto the system in order to have an onion-like security defense. While this approach leads to the stated cost

and state-space problems when done globally, the idea can be covered in heuristic approaches by iterating through the defense method types.

Another idea is to group security requirements according to viewpoints [11], e.g., according to the user's view on the system. This approach is highly dependent on the input data model. Therefore, this approach must be integrated into a method suite where the risk analysis produces the according to output. If so, the approach allows, in principle, the same heuristics as highest impact/likelihood first.

We have not found an approach that talks about the influence of attack paths with only one element. That *single risks* are either single points of failure (using the reliability language) or preparation attacks. The first case is essential to solve since these attack paths have only one step to accomplish. For the latter case, prioritizing single risks mitigate parts of several attack paths. This prioritization reduces the overall effort for risk treatment.

b) Heuristic Layers: We propose to use a combined approach as a heuristic. We prioritize single risks before tackling the highest impact first. Also, we state to use the defense-in-depth idea to assign methods from as many control types as possible to result in a layered security defense model.

The prioritization allows the extension to varying levels of detail. Possible additional layers are regarding specific impacts, e.g., safety, or likelihood impacts, e.g., certain attack surfaces, time, or knowledge level first.

C. Input Data Composition

The predecessor step of risk treatment is risk analysis of the decomposed system. An example would be separate risk analysis for each function, and the component in a function-oriented development environment.

Those different risk analysis results need to be composed and prepared for risk treatment in a single model per component. Good preparation allows to allocate defense methods on different abstraction levels and directly test for by-catch in other system parts. In the case of hardware isolation, there should be a grouping according to the units of isolation, e.g., different CPUs or existing virtualization. Otherwise, defense methods' influence or dependencies cannot be determined appropriately. Including the communication paths enables the validation of the defense methods between different communication partners.

Risk treatment according to ISO/SAE 2134 allows excluding those risks or even targets of evaluation whether the risk is below a defined threshold. Nevertheless, defense method assignment on the complete set of risks enables tracing the by-catch of excluded risks and evaluating their resulting risk level. This approach might reveal that even those risks are mitigated onto a deficient level or even entirely and by that follow the rule that someone will carry out every threat [15]. Therefore, the method assignment in our approach takes place on the relevant risks while a final by-catch test allows to have a complete overview over the remaining risk levels.

Depending on the risk analysis method, there are different ways to prepare the input data. Attack trees in the risk analysis lead to a forest of attack trees (one for each threat). The composition of those forests from the different evaluation targets leads to an extensive set of attack trees. Those directly allow deriving cut sets which can be weighted according to the impacts, e.g., attack potential, damage potential, risk. Prioritizing the weights in the assignment step allows efficient allocation of defense methods to the different threats. Depending on the defense methods' influence, this assignment reduces the weights or deletes elements from the cut sets.

Integrated risk analysis approaches have pre-defined data models that might not allow deriving attack trees and using their cut sets to allocate defense methods. In this case, look-ups and filters on the input data, e.g., for risks with a high safety impact, determine the weights. This method might seem more complicated but is automatable in most cases.

D. Approach

Following the presented taxonomy, the approach is twofold. The first step is regarding the general conditions and therefore demands necessary to fulfill. The second step accomplishes the system-related requirements.

a) *General Conditions*: Security requirements from the general conditions have to be applied to all security-relevant items. Therefore, they need no heuristic approach. They can be either applied by hand or by formally defining the rules for assignment, e.g., in a logical language.

General conditions may lead to the assignment of defense method categories (e.g., network segmentation, intrusion detection) or detailed methods (e.g., whitelist firewall on each Ethernet node).

b) *System-related requirements*: For the system-related defense methods algorithm 1 uses a set of *risks* as input. Each risk (line 1) is a set consisting of the *threat*, the *attack path*, the relation to the *item element* and applied defense methods (*applied_means*). The set *applied_means* may not be empty at the beginning since in former development steps already defense methods (e.g., structural network segmentation) may be assigned.

A risk's value is determined by the function *CRITICALITY(risk)* (algorithm 3) which combines the impact functions *DAMAGE_POTENTIAL(risk)* and *REQUIRED_ATTACK_POTENTIAL(risk)*. Those provide the impact level as well as the feasibility test for the defense methods. Please note, that for reasons of space those functions are not defined in detail in the algorithm.

The idea of the algorithm is to assign defense methods to the set of relevant risks R_{rel} as long as their risk value is above the defined threshold *criticality_threshold* (e.g., low).

In the first loop (line 5-8), all risks without an assigned attack path are evaluated. Those are single points of failure or preparation attacks. After they are mitigated, the remaining risks are evaluated in descending order of their impact (second loop, line 10-13).

Function *ASSIGN* (algorithm 2) shows the assignment of defense methods. As long as the risk under consideration is not below the threshold it tries to find a defense method *dm* and assigns it to the risk (loop line 5-23). Through the impact functions (line 10-13) the algorithm tests whether feasibility and properties of the method fit the risk. Therefore, it also takes dependencies and the technology into account. In this step, methods with the highest impact on the risk value are assigned. Other heuristic layers lead to new iteration conditions which can be easily integrated.

To find a defense-in-depth solution, which means using all control categories, the algorithm iterates over the structural, technical, and functional defense methods (line 5, 20, 22) until the risk is mitigated or no possible method is available (*fail* = 3). Other possibilities would be to use as many methods as possible from each category or until the impact is below a defined threshold. Both possibilities have a high probability that only structural defense methods are assigned which is against the defense-in-depth onion model.

In the case where a risk cannot be mitigated, the threat remains in the resulting risk value (line 24). Depending on the type of damage (Safety vs. Financial), other possibilities have to be found for mitigation, e.g., change deployment, add isolation. Therefore, the algorithm leaves this risk for dealing otherwise with it and excludes the risk from the set R_{rel} . This problem should be a corner case since the defense method catalogs provide a variety of possibilities. Nevertheless, such situations are also possible in non-automatic risk treatment procedures.

After each successful assignment, the function tries to assign the defense method to other risks (line 16-18). The assignment test in the impact functions adhere (depending on the defense method) to the units of isolation of the component. For example, a run-time protection mechanism on a virtual machine is only applied to other functions on the virtual machine. This component global assignment applies the by-catch test not only on attack paths but also on component level.

Before mitigating the next risk, the algorithm updates the set of relevant risks R_{rel} .

c) *Trade-offs*: There might be a situation where defense methods with equal impact are possible to assign. The algorithm currently uses the first method found and assigns it to the threat. Without considering costs, other defense methods could be annotated as possible substitutions, leaving the option for the security engineer to swap them.

Defense methods may have several different variants. The algorithm uses the variant with the highest impact. Therefore, one trade-off point is to mark those variant points and annotate the different impacts. The security engineer validates the results and adjusts the method variant to the preferable one. We follow the approach to have a more secure variant than less instead.

Algorithm 1 Handle risks against road user with system-related requirements**Require:**

- 1: risks as array of threat \times attack_path \times Item_element \times applied_means \triangleright at start, for all risks: applied_methods can be $= \emptyset$
- 2: criticality_threshold
- 3: means \triangleright all available means to possibly increase security

Ensure: $\mathbb{R}_{rel} = \emptyset$ \triangleright relevant risks

- 4: $\mathbb{R}_{rel} \leftarrow$ all elements r of risk with $CRITICALITY(r) > criticality_threshold$
- 5: **while** risk $\in \mathbb{R}_{rel}$ with empty attack path exists **do**
- 6: ASSIGN($\mathbb{R}_{rel}, current_risk, criticality_threshold, means$)
- 7: $\mathbb{R}_{rel} \leftarrow$ all elements r of risk with $CRITICALITY(r) > criticality_threshold$
- 8: **end while**
- 9: **repeat**
- 10: $current_risk \leftarrow$ element of \mathbb{R}_{rel} with maximal DAMAGE_POTENTIAL($current_risk$)
- 11: ASSIGN($\mathbb{R}_{rel}, current_risk, criticality_threshold, means$)
- 12: $\mathbb{R}_{rel} \leftarrow$ all elements r of risk with $CRITICALITY(r) > criticality_threshold$
- 13: **until** $\mathbb{R}_{rel} = \emptyset$

Algorithm 2 Assign defense methods to the currently processed risk.

- 1: **procedure** ASSIGN($ref \mathbb{R}, ref current_risk, criticality_threshold, means$)
- 2: **const** means_category \leftarrow [structural, technical, functional]
- 3: $i \leftarrow 0$
- 4: $fail \leftarrow 0$
- 5: **while** $CRITICALITY(current_risk) \geq criticality_threshold$ & $fail \neq 3$ **do**
- 6: $highest_impact \leftarrow 0$
- 7: $highest_impact_mean \leftarrow none$
- 8: **for all** $dm \in means$ **do** \triangleright find mean with highest impact
- 9: **if** $CATEGORY(dm) = means_category[i]$ & $dm \notin current_risk.applied_means$ **then**
- 10: **if** $CRITICALITY(current_risk \text{ with } dm \text{ applied}) > highest_impact$ **then**
- 11: $highest_impact_mean \leftarrow dm$
- 12: $highest_impact \leftarrow CRITICALITY(current_risk \text{ with } dm \text{ applied})$
- 13: **end if**
- 14: **end if**
- 15: **end for**
- 16: **if** $highest_impact_mean \neq none$ **then**
- 17: Apply $highest_impact_mean$ to $current_risk$
- 18: Apply $highest_impact_mean$ to all elements of \mathbb{R} where it can be applied
- 19: **else**
- 20: $fail \leftarrow fail + 1$
- 21: **end if**
- 22: $i \leftarrow (i + 1) \bmod 3$
- 23: **end while**
- 24: **if** $fail = 3$ & $CRITICALITY(current_risk) \geq criticality_threshold$ **then**
- 25: Deal otherwise with $current_risk$
- 26: $\mathbb{R} \leftarrow \mathbb{R} - current_risk$
- 27: **end if**
- 28: **end procedure**

Algorithm 3 Determine updated risk from damage potential and required attack potential.

- 1: **function** CRITICALITY(risk)
- 2: **return** DAMAGE_POTENTIAL(risk) \cdot REQUIRED_ATTACK_POTENTIAL(risk)
- 3: **end function**

V. DISCUSSION

The presented approach is just a starting point for further possible extensions. Therefore, there are some limitations. Also, there are further possible trade-offs when transferring the idea to other system types.

A. Limitations

Automatic assignments of defense methods always have the probability of misjudgment [18]. This limitation also accounts for the current version of the presented approach. Therefore, the result needs to be manually validated.

The current approach excludes compatibility validation of the assigned defense methods for interface security between different security partners. This limitation can lead to inconsistencies between communication partners. In vehicle projects, there are few different defense methods regarding communication interfaces. Also, vehicle software functionalities distribute onto different components. Therefore, a corner case should be items where the communication partner is another functionality on a different component not providing the demanded defense method.

Without an architectural verification method in place, the assumptions regarding communication partners' security risk levels, e.g., secure back-end, need to be verified manually. This verification includes several steps:

- Whether a risk analysis for the communication partner exists.
- If the risks targeted by the assumption are covered
- If the risk treatment properly mitigates these risks.

The risk treatment step results are typically imported into an integrated modeling tool that illustrates the complete vehicle architecture. This process reveals inconsistencies in defense methods between communication partners. Also, assumptions regarding the security level of a communication partner are verifiable in this model.

B. Optimizations

The algorithm in the current form prefers methods that have the highest impact on all threatened security goals. This priority led to the assignment of defense methods that impact more than one security goal before the others. Typically it is cheaper to implement one costly defense method, which impacts more security goals, than to implement several defense methods which target fewer security goals. Nevertheless, this preference might lead to higher costs in some cases. Including the implementation costs into the assignment leads to a higher state space and a more optimal solution regarding the costs.

On the other hand, it is costly and challenging to achieve the implementation costs of the defense methods. Since, for some defense methods, the costs are hardware and implementation-dependent, the costs change for each hardware type. Average costs might prevent this problem while still optimizing the assignment. At least for each vehicle project, the costs have to be newly evaluated when the defense method catalog is updated.

ISO/SAE 21434 clearly defines impacts only to the road user. Therefore, the current version of the algorithm mitigates only threats with an impact on the road user. Typically, threats have an impact on both stakeholders to a varying degree. Only a few threats remain untreated by targeting the threats with impact against the road user. Those are only regarding the OEM, or the impact against the road user is less than the threshold while the impact against the OEM is above the threshold. Untreated threats might benefit from assigned defense methods. A global by-catch test on the complete list of threats reveals such situations. Nevertheless, this is a possible point to optimize the system setup. A second threshold for the OEM impact allows a second run of the algorithm over those threats. Combined with a cost limit OEM-related threats could be mitigated to a certain degree.

For other system types, not every risk needs to be mitigated, e.g., in IT security. In those cases, other trade-offs in the risk treatment are possible, for example: by weighting the attack potential categories or damage potential categories. Other possibilities are to mitigate just certain threat classes.

VI. CONCLUSION AND FUTURE WORK

The presented paper aims to make a step toward an efficient and traceable risk treatment method for automotive development projects, transferable to other system settings. At first, we defined an adjustable taxonomy for the terminology of security requirements. By that, we address the problem of differing terminology in practice and literature. The taxonomy enables building up a database like defense method catalog. Regular updates of the catalog incorporate newly revealed threats.

Our approach for risk treatment has high flexibility regarding the input data. We use general information from risk analysis instead of detailed data models. With this approach, we trade practicability against fine-granular defense method assignment. We know that this approach limits the verification possibility on the architectural level and leaves this for the pen-testing process step. Nevertheless, we think that a pragmatic approach that might overestimate the risks in corner cases with less effort is cheaper in the overall process.

Also, the approach can incorporate different origins of defense methods. Normative and organizational security demands need to be adhered to in the development. Otherwise, the system is infeasible from those points of view. Therefore, we differentiate between those security demands and risks revealed in the risk analysis step. On the one hand, this distinction allows tracing the source of assigned defense methods. On the other hand, this enables trade-offs in the assignment of system-related defense methods.

Efficient risk treatment demands a particular structure in the assignment process. Therefore, we mitigate threats without attack paths first. Single threats are either single-points-of-failure or preparation attacks. The former need mitigations for a high-security defense. The latter have a high by-catch rate since they directly cut attack paths. After that, we prioritize the attack path with the highest impact against the road user.

The decision for the highest impact first, as well as other trade-off points, is easily adjustable. Our approach basis is a heuristic that allows for different trade-offs and cost inclusion.

Especially the incorporation of costs is part of significant future work. Although the costs of the different defense methods are challenging to obtain, they allow optimization and trade-offs in method assignment.

Other future work is regarding the discussed limitations. We aim for a method to incorporate the compatibility between the different components and verify assumptions. This method demands a verification approach on the architectural level refined throughout the complete development process. For that, preliminary work exists [19] [20] [21], which we now want to develop further.

An open question is regarding the incorporation of the negative influence of defense methods on other functionalities [18]. This influence is crucial, especially in optimized and resource constraint systems like the automotive industry. One idea would be to include this in the mentioned verification approach. If the model allows behavioral verification of all system parts, the influence of defense methods can be directly composed into this model.

REFERENCES

- [1] ISO, "ISO/SAE 21434:2021 Road vehicles – Cybersecurity engineering," 2021.
- [2] C. Jakobs, B. Naumann, M. Werner, K. Schmidt, J. Eichler, and H. Heskamp, "Streamlining Security Relevance Analysis According to ISO 21434," in *Proceedings of the 5th International Conference on Networking, Information Systems & Security (NISS'22)*. IEEE, 2022, to appear.
- [3] U. Nations, "Proposal for a New UN Regulation on Uniform Provisions Concerning the Approval of Vehicles with Regards to Cyber Security and Cyber Security Management System (UN Regulation No. 155)," 2020.
- [4] C. Jouvray, A. Kung, M. Sall, A. Fuchs, S. Gürgens, R. Rieke, Y. Roudier, and B. Weyl, "EVITA Deliverable D3. 1: Security and trust model," Tech. Rep. 3.1, 2009.
- [5] C. P. Pfleeger and S. L. Pfleeger, *Security in Computing*, 3rd ed. Prentice Hall, 2003. ISBN 978-0-13-035548-5
- [6] L. Chung, Ed., *Non-Functional Requirements in Software Engineering*, ser. The Kluwer International Series in Software Engineering. Kluwer Academic, 1999. ISBN 978-0-7923-8666-7
- [7] A. Akhunzada, E. Ahmed, A. Gani, M. K. Khan, M. Imran, and S. Guizani, "Securing Software Defined Networks: Taxonomy, Requirements, and Open Issues," *IEEE Communications Magazine*, vol. 53, no. 4, pp. 36–44, 2015. doi: 10.1109/MCOM.2015.7081073
- [8] M. Ahmadvand, A. Pretschner, and F. Kelbert, "A Taxonomy of Software Integrity Protection Techniques," in *Advances in Computers*. Elsevier, 2019, vol. 112, pp. 413–486.
- [9] State Administration for Market Regulation; Standardization Administration of the People's Republic of China., "Technical requirements and test methods for cybersecurity of on-board information interactive system (GB/T 40856-2021)," 2021.
- [10] "Recommended Practice: Improving Industrial Control System Cybersecurity with Defense-in-Depth Strategies," 2016-09. [Online]. Available: https://www.cisa.gov/uscert/sites/default/files/recommended_practices/NCCIC_ICS-CERT_Defense_in_Depth_2016_S508C.pdf
- [11] B. Weyl, M. Wolf, F. Zweers, T. Gendrullis, M. S. Idrees, Y. Roudier, H. Schweppe, H. Platzdasch, R. El Khayari, O. Henniger *et al.*, "EVITA Deliverable D3. 2: Secure On-board Architecture Specification," 2011.
- [12] C. Irvine and T. Levin, "Toward a Taxonomy and Costing Method for Security Services," in *Proceedings 15th Annual Computer Security Applications Conference (ACSAC'99)*. IEEE Comput. Soc., 1999. doi: 10.1109/CSAC.1999.816026 pp. 183–188.
- [13] K. Lee, Y. Lee, H. Lee, and K. Yim, "A Brief Review on JTAG Security, year=2016, pages=486-490, doi=10.1109/IMIS.2016.102," in *2016 10th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS)*.
- [14] D. Angermeier, K. Beilke, G. Hansch, and J. Eichler, "Modeling Security Risk Assessments," p. 14, 2019. doi: 10.13154/294-6670
- [15] A. Ruddle, D. Ward, B. Weyl, S. Idrees, Y. Roudier, M. Friedewald, T. Leimbach, A. Fuchs, S. Gürgens, O. Henniger *et al.*, "EVITA Deliverable D2.3: Security Requirements for Automotive on-Board Networks Based on Dark-Side Scenarios," 2009.
- [16] B. Schneier, "Academic: Attack Trees - Schneier on Security," 1999. [Online]. Available: https://www.schneier.com/academic/archives/1999/12/attack_trees.html
- [17] S. Mauw and M. Oostdijk, "Foundations of Attack Trees," in *Information Security and Cryptology - ICISC 2005*, ser. Lecture Notes in Computer Science, D. H. Won and S. Kim, Eds. Springer Berlin Heidelberg, 2006, vol. 3935, pp. 186–198.
- [18] G. Hansch, "Automating Security Risk and Requirements Management for Cyber-Physical Systems," 2020.
- [19] C. Jakobs, M. Werner, K. Schmidt, and G. Hansch, "Following the White Rabbit: Integrity Verification Based on Risk Analysis Results," in *Computer Science in Cars Symposium*. ACM, 2021. doi: 10.1145/3488904.3493377
- [20] C. Jakobs, M. Werner, and P. Tröger, "Dynamic Composition of Cyber-Physical Systems," in *52th Hawaii International Conference on System Sciences (HICSS)*, 2019. doi: 10.24251/HICSS.2019.869
- [21] C. Jakobs, B. Naumann, M. Werner, and K. Schmidt, "Verification of Integrity in Vehicle Architectures," in *Proceedings of the 3rd International Conference on Networking, Information Systems & Security*. ACM, 2020. doi: 10.1145/3386723.3387883

Universal Key to Authentication Authority with Human-Computable OTP Generator

Sławomir Matelski
 Research and Development
 INTELCO LLC
 Lodz, Poland
 s.matelski@intelco.pl

Abstract—The subject of this paper is an enhanced alternative to the Multi-Factor Authentication (MFA) methods. The improvement lies in the elimination of any supplementary gadgets/devices or theft-sensitive biometric data, by substituting it with direct human-computer authentication optionally supplemented by cognitive biometric. This approach remains secure also in untrusted systems or environments. It allows only one secret as a universal private key for all obtainable online accounts. However, the features of this new solution pretend it to be used by the Authentication Authority with the Single-Sign-On (SSO) method of identity and access management, rather than for individual services. This secret key is used by our innovative challenge-response protocol for human-generated One-Time Passwords (OTP) based on a hard lattice problem with noise introduced by our new method which we call Learning with Options (LWO). This secret has the form of an outline like a kind of handwritten autograph, designed in invisible ink. The password generation process requires following such an invisible contour, similar to a manual autograph, and it can also be done offline on paper documents with an acceptable level of security and usability meeting the requirements for post-quantum symmetric cyphers and commercial implementation also in the field of IoT.

Index Terms—authentication, lattice, OTP, secret key.

I. INTRODUCTION

DUE TO the growing threat of cyber attacks, multi-factor authentication (MFA) or the two-step verification has recently become a cybersecurity standard.

Step 1 - comprises entering a user ID and static password. For security reasons, it is recommended to use different passwords for each online account. As a result, users often adopt insecure password practices (e.g., reuse or weak password) or they have to frequently reset their passwords. Blocki et al. introduced in [7] an innovative Human-Computable Passwords (HCP) scheme, which ensures that even if an adversary has seen one-hundred of the user's passwords still has high uncertainty about remaining passwords. The disadvantage of their scheme is the need to memorize dozens of pictures, mapping to numbers with the help of associated mnemonics.

In such an HCP scheme the user reconstructs each of his passwords by computing the response to a public challenge, by performing simple mathematical operations i.e. addition modulo 10. A similar approach to the idea of password computing is used by our iChip protocol inspired by the topography of electronic microchips and handwriting (Fig. 1).

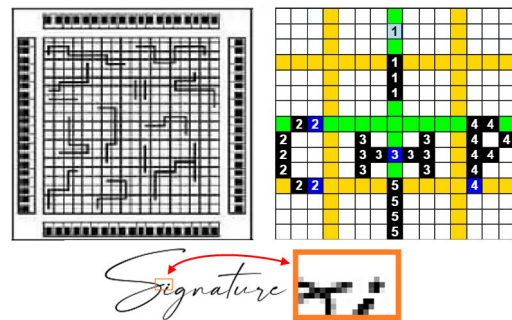


Fig. 1. Topography of: microchip, iChip, handwriting.

It requires much less effort to remember the secret in the form of only one (but detailed) picture, and only half the time for authentication. As we show in Section VI, it guarantees a safe generation of many thousands of such passwords. What follows, it can be used as an OTP generator as well.

Step 2 - requires usually an additional electronic device (using the same device in both steps may not be safe), that uses an embedded one-time password (OTP) generator or biometrics. In such cases, the OTP is entered into the verification system automatically, e.g., from a smartcard or an IoT device, or by the user after being read from the screen of a token or personal smartphone via SMS or special application. Unfortunately, this solution does not ensure that the device is being used by its owner; it must be always available, and can be stolen, lost, damaged or cloned. Biometric methods are an alternative, but these can be relatively easily cheated by *replay attack* using snooped biometric data, and with help of machine learning or AI algorithm if necessary [18].

The MFA obviously requires more time than entering a regular static password. Therefore, a human generated OTP protocol with comparable authentication time and sufficient security, eliminating the long list of drawbacks mentioned above, stands a chance of mass user acceptance.

Many attempts to achieve this goal have been made for over 30 years since Matsumoto's first publication [1] in 1991, but only two protocols have been commercially implemented: HB presented by Hopper and Blum in [2] and GrIDSure (GS) presented in [19]. We will show further that our iChip scheme has security properties better than HB and usability close to GrIDSure, while eliminating their drawbacks.

- The HB is based on the Learning Parity with Noise (LPN) method, which ensures a high level of security, but the time of ca. 668 sec. needed for authentication by a human is too long to be acceptable. Nevertheless, the properties of this protocol or later improved variants (HB+, HB#) are well suited to applicate in resource-constrained devices, such as Internet of Things (IoT) devices or RFID.

- The GrIDSure scheme has exactly the opposite properties confirmed in [19]: the high usability level and very low level of safety, as only 3 samples of challenge-response pairs are sufficient to reveal the secret. In addition, the entropy of this scheme is also low, as detailed research has shown, that users choose secret patterns that are easy to remember and frequently reused, so its scheme is highly vulnerable to dictionary attacks, as the choice is very limited due to the small grid and the small number of secret objects. The only effective improvement proposed in [20] is the use of a few secrets switched by the Out-of-Band (OOB) channel, but that requires the employment of an additional device that we intend to eliminate.

The iChip has similar usability properties to GrIDSure as the secret pattern of cells in the grid is employed by both schemes. However, the similarity is noticeable only in the so-called generator block. The most significant difference lies in the extraordinary mapping method used in iChip, which makes a huge difference in the key space ($3e+5$ vs $3e+154$), and provides many thousands of times greater resistance against peeping attacks than GS. The conclusions about low practical entropy of GS do not apply to the iChip as getting all the easy-to-remember keys from such a huge key space is a task with a difficulty near to brute-force, which is not feasible for current supercomputers.

As mentioned above, the iChip is applicable also in step 1 of the MFA as one universal secret key to the creation of multiple original static passwords for each online account. However, the first step of MFA is redundant in this case as it relates to the same secret as the OTP generator. On the other hand, instead of the 1st step, we propose a discreet introduction of the 2nd factor in the form of cognitive memory using proposed in Section III-E, and Single Sign-On (SSO) method based on the OAuth2.0 protocol [23] under control of authentication server (as an Authenticating Authority) that owns the user identities and credentials, including the iChip secret key or container with multiple pairs of challenges and hashed OTPs.

The contributions of this work are: the challenge-response cryptographic protocol, based on lattice problem with noise, introduced by our Learning with Options (LWO) method as a more effective new variant of the LPN method of easy OTP computation by a human; a graphical interface for the implementation of that protocol, which allows the user to create his secret in the form of an easy-to-remember image, and a special wizard to compose it; both well-proven in usability and security study discussed after the presentation of mathematical rules, illustrated by examples of the iChip core and its TurboChip overlays; further protocol enhancement against active attacks and by cognitive memory usage.

The completed implementation can be tested in an interactive demo or viewed in a short film, either as a professional tutorial or an alternative version made by children participating in the research process; both available online [25]. It is much more effective to understand than a mathematical description.

II. LEARNING WITH OPTIONS METHOD AS THE MAIN STRENGTH OF THE PROTOCOL

An important element of the iChip scheme is the implementation of the Learning with Rounding (LWR) method, which is an LPN variant of worst-case hard lattice problem included in the lattice-based cryptography. The implementation of core LPN or Learning with Errors (LWE) methods increase the security of any protocol; however, the degree of usability is reduced, and authentication requires much more time, as the user has to perform additional protocol rounds to compensate for rounds lost to incorrect responses due to reduced resistance to random attacks. In contrast, the LWR and described below LWO methods requires only correct responses. The iChip uses Equation 1 in Section III-A, as its base function which satisfies the criteria of the LWR method of deterministic rounding by $x \bmod p$, where $p = 10$ is admittedly too small to effectively introduce noise, but convenient for human computation. This function is a node for the various protocol variants and for our proposed LWO method of introducing noise, which is far more efficient.

III. THE ICHIP AS AN OTP GENERATOR

The iChip is a challenge-response protocol to authenticate the user to the verifier using the shared secret, where the user has to answer the challenge generated by the verifier (server). The way the iChip scheme worked was inspired by the image of the photolithographic mask used to create conductive paths on the surface of PCBs (Printed Circuit Board) or ICs (Integrated Circuits) like shown in Fig.1. The user composes his secret by designing such a layout in a special wizard by drawing a map of blocks B of masking elements as paths conducting the digital signal from input to output; provided from the generator block. These paths will determine the change in value from V_{inp} at the input to V_{out} at the output and define their properties and mutual logical relations. This layer consists of $n \times n$ fields and is represented by the C matrix, containing $n \times n$ cells.

The user specifies his secret key S by specifying a list of b blocks that occupy the fields selected by him from the C matrix, and specifies the block elements that act as input or output. For a short and easy explanation, we will use the example of the secret key illustrated in Fig. 2 or Fig. 4 as an iChip layout and the matrix coordinates of the input and output elements encoded hexadecimal in the associated table, while for the description of the protocol, we will use the Python convention. The C matrix is a set of n^2 random values generated by verifier as $C = [[V_{1,1}, V_{1,2}, \dots, V_{1,n}], \dots [V_{n,1}, V_{n,2}, \dots, V_{n,n}]]$. Each i -th element of block $B[i] = [y_i, x_i, z_i]$ is defined by 3 parameters: row y and column x as field coordinates (y, x) in matrix C , and parameter z defining its state: $z = \{I, O\}$, where: $I = Input$, $O = Output$.

We use also an alternative compact notation of block elements as: $B_j^z[i] = B_j^z[(y_i, x_i)]$. Each block B_j is a list of such elements, divided into two segments for inputs and outputs: $B = [B^I, B^O] = [B[1], B[2], \dots, B[k]]$. A list of b blocks B_j is included in the secret $S = [B_0, B_1, \dots, B_{b-1}]$, where $0 \leq j < b$. The parameters of the algorithm are denoted by four positive integers $N, L, b, k \in \mathbb{N}$, where:

- chip size (the matrices describing both private part of the key and the challenge matrix have size $N = n \times n$);
- parameter describing OTP length, $L \leq 10$;
- maximal number of blocks, for the sake of clarity and memorability we restrict $1 \leq b \leq 10$;
- maximal block length $k \leq 10$;

A. Generating OTP

$G = B_0$ is the first of these blocks in key S , and it is called a generator because it does not contain inputs and the values $V_G = C[G]$ from all its $L = |B_0|$ output elements are mapped by the remaining blocks. The user has to remember the position of all blocks and their order in the S . The verifier generates a challenge matrix C of N random digits. To generate the OTP, the user has to collate the C matrix with the secret key S and calculate all OTP digits, one at each i -th of $L = |\text{OTP}|$ rounds of the protocol in the following 3 steps:

- 1) Read the V_{inp}^i value of the $G[i]$ element in C at position (y_i, x_i) : $V_{inp}^i = C[G[(y_i, x_i)]]$
- 2) Starting from j -th block (where $j = 1$ in the 1st round), search input elements ($z = \text{Input}$) of j -th block for the coordinates (y_i, x_i) such that $V_{inp}^i = C[B_j^I[(y_i, x_i)]]$. If no such coordinates are found in the j -th block, move to the subsequent block. By $j = \phi$ denote the index of the *current block* (ϕ) in which the searched so-called *target input* (ψ) has been found first and let

$$V_{out}^i = C[B_\phi[y, x, z = \text{Output}]].$$

If the search fails for all $j < b$: let $V_{out}^i = V_{inp}^i$.

- 3) The i -th digit of the OTP you will get as

$$\text{OTP}[i] = (V_{inp}^i + V_{out}^i) \bmod 10 \quad (1)$$

To avoid overloading the first blocks, it is recommended to resume the search for V_{inp}^i from the block next to the last searched. For additional security, the following three exceptions/rules (*I, *O, *Θ) have been added to the 2nd step of the algorithm; these significantly increase the resistance of the iChip protocol against passive attacks with a statistical algorithm or Gaussian Elimination. For their consideration let (y_i, x_i) be the coordinates on which the *target input* ψ in B such that $C[B_\phi[\psi]] = V_{inp}^i$ was found first in the challenge matrix.

However, first, we will present a simple example illustrated in Fig. 2 to explain the principle of calculating the OTP without the exceptions mentioned above. Alternatively, it is recommended to watch the short video tutorial [25].

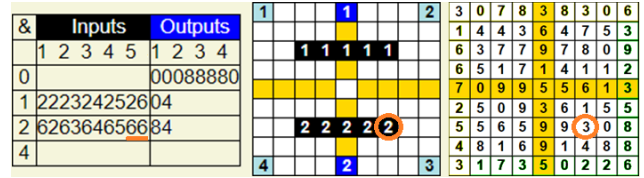


Fig. 2. An example of a secret: block input elements given as black fields and output as blue or light blue fields. The positions of all input and output elements are hexadecimal encoded in the associated table. The first column (&) contains the index of each block. On the right: The challenge matrix.

There are generator block 0 containing 4 light blue cells in the matrix corners and two mapping blocks labeled by their index (1 or 2) in the example above (Fig. 2). In the 1st round, we read the value $V_{inp}^1 = 3$ from the 1st element of generator block at position (0, 0).

We look for this value sequentially in all mapping blocks from 1 to 2. The first occurrence of this value is in the last element of block 2, i.e. $B_2^I[5]$ in cell (6, 6), which is the *target input* $\psi = 5$ in the *current block* $\phi = 2$. Now, we read a value of the output element of this block, which is in cell (8, 4), hence $V_{out}^1 = C[8, 4] = 5$. The 1st round ends with a calculation of the 1st OTP digit according to Equation 1 as:

$$\begin{aligned} \text{OTP}[1] &= (V_{inp}^1 + V_{out}^1) \bmod 10 = \\ &= (C[G[0]] + C[B_2^O[1]]) \bmod 10 = \\ &= (C[0, 0] + C[8, 4]) \bmod 10 = \\ &= (3 + 5) \bmod 10 = 8. \end{aligned}$$

EXTRA RULES / EXCEPTIONS

*I) Let V_{inp}^i be the sum of all input elements of the current block, from ψ to $\psi + n$, where $\psi + n \leq |B_\phi^I|$ and $n \leq 2$:

$$V_{inp}^i = \left(\sum_{k=0}^{n} C[B_\phi[\psi + k]] \right) \bmod q \quad (2)$$

This introduces non-linearity to cryptanalysis and protection against Gaussian Elimination, as the number of arguments in Equation 2 varies randomly in each challenge. Depending on the variant of *I, the q modulus can be 10 or omitted as default.

*O) If the *current block* B_ϕ contains more than one output element $|B_\phi^O| > 1$, then randomly choose one of them as $V_{out}^i = C[B_\phi^O[\text{randrange}(1, |B_\phi^O|)]]$. This is the case of using the LWO method illustrated by Fig. 3: Block 2 with fields labeled by 2 has two outputs/options at positions (3, 1) and (3, 3). If the value searched for is found in this block, then the user has to choose one of these two options at random.

*Θ) If the *current block* B_ϕ has no output $|B_\phi^O| = 0$, then we use the next input instead: $V_{out}^i = C[B_\phi[(\psi + 1) \bmod |B_\phi|]]$.

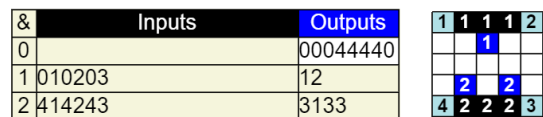


Fig. 3. An example of secret with exception *O.

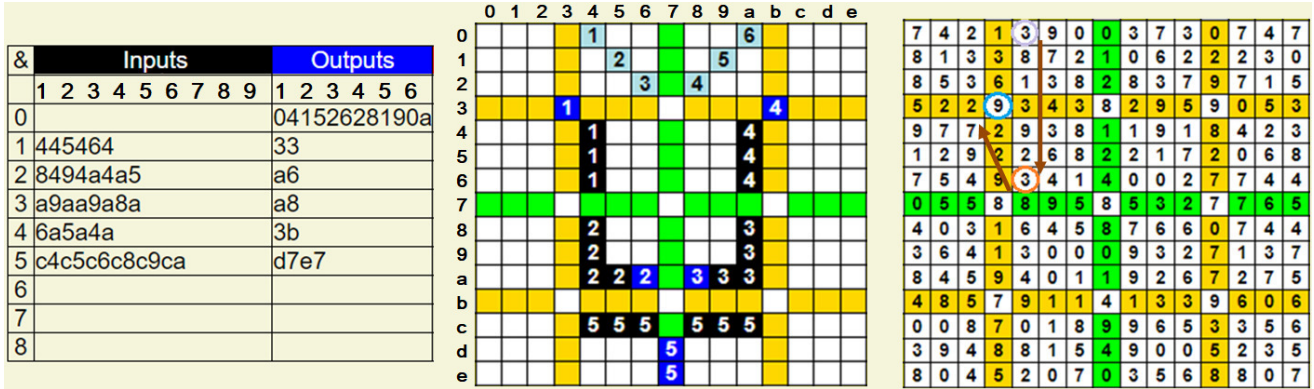


Fig. 4. An example of a secret defined by the user and challenge matrix with a schema for determining the 1st digit of OTP.

B. Advanced Example

Based on Fig. 4 we will compute the 6-digit OTP as follows: The generator block contains $V_G = C[G] = [3, 7, 8, 8, 6, 3]$. The 1st element $V_G[1]$ at position (0, 4) has a value of 3. When looking for it sequentially in blocks 1 to 6, it can be found in the 3rd input element of the 1st block $B_1^I[3]$ at position (6, 4), (marked in the red ring as a target input ψ); the output element $B_1^O[1]$ of this block is in cell (3, 3) with a value of 9. The 1st digit of OTP is calculated according to Eq. 1 as $OTP[1] = (3 + 9) \bmod 10 = 2$.

$V_G[2]$ at position (1, 5) has a value of 7, which is also in $B_4^I[2]$ at position (5, a) and the output element $B_4^O[1]$ has a value of 9. Since $\psi = 2 < |B_4^I| = 3$, then due to rule *I: $V_{inp}^2 = C[B_4^I[2]] + C[B_4^I[3]] = 7 + 1 = 8$. Now, according to Eq. 1 we can calculate: $OTP[2] = (8 + 9) \bmod 10 = 7$.

$V_G[3] = 8$ appears in $B_5^I[3]$ at position (c, 6), but this block has 2 output elements $B_5^O[1]$ in cell (d, 7) and $B_5^O[2]$ in cell (e, 7). Therefore due to exception *O, we can choose any of them; assuming we choose the first with value of 4, $OTP[3] = (8 + 4) \bmod 10 = 2$.

$V_G[4] = 8$, hence this round is similar to the previous one, but now, we use the second output $B_5^O[2]$ in cell (e, 7) for our calculations: $OTP[4] = (8 + 0) \bmod 10 = 8$.

$V_G[5] = 6$ appears in $B_7^I[1]$, but this block has 4 inputs, therefore we add three of them to $V_{out}^5 = 1$, hence: $OTP[5] = 6 + 3 + 4 + 1 \bmod 10 = 4$. Entire OTP = [2, 7, 2, 8, 4, 2].

C. TurboChip overlays for the iChip protocol

For a radical reduction of an authentication time, we have developed two variants of TurboChip overlays for the iChip scheme. They only uses 1 round of the base iChip protocol and only needs 1 element in the generator block. In the example below illustrated in Fig.5: Since $V_G = [4]$, then $OTP[1] = (4 + 2) \bmod 10 = 6$. However, we keep this value a secret as $V = 6$ and use it according to FlexiChip or ClickChip. For FlexiChip, we calculate each i -th OTP number as $OTP[i] = (V + C[B_\phi^I[\psi + i]]) \bmod 10$; If $|B_\phi^I| < \psi + i$ then continue in the next block. In this example: $\phi = 1; \psi = 3$, but $|B_1^I| < 3+1$, hence $OTP[1] = V + C[B_2^I[1]] \bmod 10 = 6 + 8 \bmod 10 = 4$; $OTP[2] = 6 + C[B_2^I[2]] \bmod 10 = 6 + 1 \bmod 10 = 7$; e.t.c.

The ClickChip generates response as matrix coordinates instead of digits. This approach requires a bit of proficiency from the user, however, it allows cutting the number of protocol rounds in half. It stands as a good trade-off for it and is explained in the example illustrated in Fig. 5. Determining of 3 matrix coordinates in the range of (-8, -8) to (8, 8), one at each i -th of 3 rounds of ClickChip protocol is as follows:

We go to the i -th element in $j = \phi + i$ block. Now, get the value of this element to calculate the distance of d_i fields, where $d_i = (V + C[B_j^I[i]]) \bmod 10$; to move the virtual pointer on a horizontal, vertical or diagonal line towards the centre of the grid. Important notes: The order of choosing these directions is random, but 1 diagonal and 1 reverse direction must be used if d_i is lower than the distance from the edge of the grid. For long blocks: If $|B_j^I| > 3$ then the counting of d_i cells starts from the element whose value is closest to d_i .

To quickly find the endpoint, move the pointer in jumps a 4 fields, with a help of the coloured background lines.

In our example, the current block $\phi = 1$, therefore, in the 1st round we go to block 2, and get the value of the 1st element in it, hence $d_1 = (6 + 8) \bmod 10 = 4$. Now, we move the pointer e.g., diagonal from a position (-1, -3) to position (3, 1). In the 2nd round $d_2 = (6 + 3) \bmod 10 = 9$, so we click the position (-3, -6). In the 3th round $d_3 = (6 + 7) \bmod 10 = 3$, but to satisfy the protocol rules, we alter the direction to the opposite $d_3 = -3$ and then click the cell (6, 6).

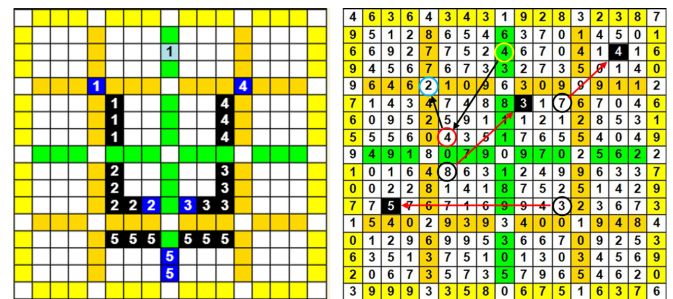


Fig. 5. Secret key and challenge matrix with the schema of the ClickChip for OTP determining protocol.

The variant with an 18x18 grid shown in Fig. 6 is more convenient for moving the pointer in jumps a³ fields. The worst-case probability of randomly hitting the correct OTP is here $p = 3/207 \cdot 2/207 \cdot 1/207 = 7e^{-7}$.

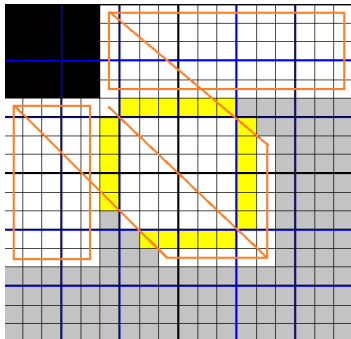


Fig. 6. ClickChip’s grid variant of 18x18 = 324 cells with 25 secret inputs marked in black and the non-clickable area marked in grey.

D. Preliminary stage against active attacks

In this stage, User U and Verifier V swap their roles, so V responds to U’s challenge. V initiates the authentication process by sending a challenge to U. Then, U randomly clicks on any field in C, and the element of the block closest to that field is used as ψ for the calculation of OTP[0]. U remembers it and sends such a challenge to V. In response, V has to calculate the OTP[0] in this same way, and then generate a new challenge, but the value of V_G[0] is set to OTP[0], which is hidden. The next 3 rounds run as usual according to ClickChip.

E. Biometrics, captcha and clock in the iChip scheme

To further strengthen the iChip protocol it is beneficial to increase the entropy of the random option selection in the LWO case by aid of a pseudorandom number generator, the result of which is entered via a cognitive biometric interface working like an OOB channel. This interface (emOTP) is based on the stimulation of emotional states by recalling the knowledge already acquired in the past and preserved in long-term memory. The great advantage of this approach is that the user is not required to remember a secret specially built for this purpose, so they can without much effort, insert a lot of such items into the user’s account profile resources in the form of catchwords or pictures, associated with its evaluation in points: +1 as positive, -1 as negative and 0 as neutral; referring to a universal question, e.g., *Do you like it?* with possible answers: *Yes, No, Neutral*. The response can be effortlessly applied to choose/address 1 of the 2 or 3 options when using the LWO method.

Increasing the range of ratings to 10 points requires changing the above question to *How much do you like it?*. Now, the response ranging from 0 to 9 allows the generator block to be completely replaced. Unfortunately, the limitation of such a generator is its inaccuracy, occurring from poor behavioural repeatability. Nevertheless, due to the LWO, the unintended

incorrect response to the emOTP challenge does not affect the response correctness. Hence, an additional protocol round to compensate for this mistake is not needed. In other cases than the LWO, such a 3 stage emOTP trigger/generator in the iChip protocol can be used as a biometric factor in the MFA. The emOTP criteria should be quick to evaluate and be personal, rather than popular and predictable. The challenge in the form of an image (as an example in Fig. 7) can work well as a captcha at the same time.

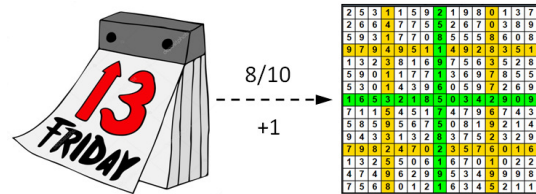


Fig. 7. An example of the emOTP challenge and its evaluation depending on the user and not on a statistical basis.

As an alternative and much simpler source of entropy for the LWO method, a random moment of reading seconds from the system clock or user’s watch can be used.

IV. ICHIP FOR A HASH-BASED SIGNATURE

To ensure that the user authorizes the correct message M (e.g. transaction conditions), and not falsified by active adversary, we propose the Human-Hashed variant of (hash-based) message authentication code (MAC). Such a HHMAC can be used offline, i.e. the previously computed SHA-256 function from M and written here as h() is hashed by iChip’s OTP. The iChip-256 variant with N=256 fields layout is the most optimal here. The challenge matrix C created by the RNG is modified by adding one bit of the hashing result H to each of the |C| = 256 elements, where $H = h(M, C)$ is the hash value of a message M and C according to the formula $C'[i] = (C[i] + H_2[i]) \text{ mod } 10$, where $0 \leq i \leq 255$. The user performs the signature by entering the OTP on the keyboard or writing OTP digits on the document containing: a blank iChip grid for global UID (optional), the challenge matrix C’, a QR code specifying the document identifier in the repository for automatic scanning and HHMAC verification.

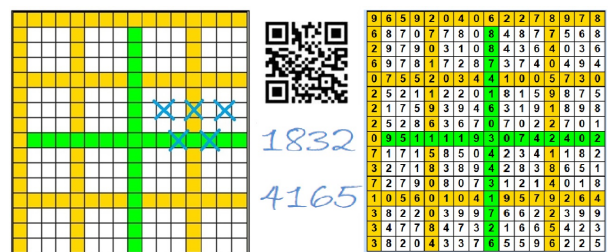


Fig. 8. A sample signature on a paper document includes: Global UID, Transaction ID, HHMAC, Challenge C’.

V. BRIEF ANALYSIS OF USABILITY

- Intelligibility

Our time-limited study only focused on a small group of children aged 8-10, assuming that the adult performance should be better, because modular addition and abstract thinking are required, which develops with age [16]. For this group, the iChip protocol was compared with that of a board game, more especially the well-known Monopoly or Jumanji, where the throws of the dice symbolize the operation of the generator block, and all the fields on the board forming the track constitute the iChip blocks, which user have to go to achieve the target field/input and finally make a decision according to the rules of the game protocol. The children took 1 standard lesson unit (45') to learn the protocol and the special wizard to design their own microchip.

- Memorizing and Rehearsing

The appropriate distribution of block elements is of major importance for entropy level and easy memorization of the entire structure of the secret. To obtain the maximum practical entropy and to make it easier to remember the secret, a suitable background image is very helpful, which can be built individually by the user or proposed by the wizard as a random structure. It is profitable to draw the secret contours in a single sequence like a short piece of text (e.g., Fig. 1) or a simple shape (e.g., Fig. 5). Additionally, since all key elements are used each time, the whole secret image can be easily remembered after 30-45 minutes of repeated authentication training attempts and frequently refreshed at the use stage.

- Authentication Time

The authentication time is proportional to the user's cognitive workload - ranges from 4 to 8 seconds (≈ 6) in each round of response, depending on the composition of the secret and the user's skill. After several searches, visual perception adapts a parallel analysis approach, i.e. the search for an ψ element with V_{inp} is not performed element-by-element, but in blocks, just like reading a text, with whole words being interpreted, rather than individual letters. Each modular addition and block search require ca. 1 sec. For the user who has to look at the keyboard to enter OTP, it will be easier, and faster, to use voice input, which is also a good source of biometric data and a 3-rd authentication factor. After introducing of TurboChip overlay, the authentication time is significantly reduced up to ≈ 15 seconds if the user becomes an experience in using it.

VI. BRIEF ANALYSIS OF SECURITY

The resistance to a random attack depends on the number of OTP digits calculated by the user. Their number L is arbitrary and depends on the needs of the authentication system, e.g., $L = 6$ like OTP in most e-banking systems. Using the ClickChip overlay slightly weakens the protocol if the attacker records user responses, as it allows to reduce the number of possible click locations from $N = 324$ down to 207 (in the worst case - Fig. 6), however, the probability of $p=7e-7$ of randomly hitting the correct OTP is still lower than for 6 decimal digits, i.e. $p=1e-6$.

The iChip's resistance to active attacks is ensured in the preliminary stage (see Section 3.4) or by the hashing and signing the authenticated message, as the HHMAC is valid only for the signed message (see Section IV).

As the challenge in the iChip protocol is generated full at random, it is fully immune to frequency analysis.

The iChip's entropy is of course lower in practical use than its key size of 512 bits, but much higher than a text password due to large number of possible fonts and their positioning on the large grid or cell order. A good example is the word *iCHIP* used in Fig 1. The number of possibilities for designing this contour is enormous, despite the use of many symmetries compared to the number of combinations that the use of lowercase and uppercase letters offer.

The resistance to brute-force and Grover's quantum algorithm is provided by NP-hard lattice problem and huge keys space (see Table 1), estimated as follows:

$$\frac{N!}{(N-L)!} \cdot \sum_{i=B}^{B+b_0} \left(\sum_{d=1}^k \binom{N-L}{d} \right) \cdot \sum_{j=E}^{E+e_0} \binom{N-L}{j} + \sum_{j=E}^{E+e_0} \frac{(N-L)!}{(N-j-L)!} \quad (3)$$

where:

$N = n \times n$ is the size of Chip's matrix, default 16 x 16

L is the number of OTP digits, default 6

$[B, B + b_0]$ = number of blocks, in the range 3 to 7

$[E, E + e_0]$ = number of input elements in block: 3 to 9

$[0, k]$ = number of output elements in block: 0 to 2 or max. 3

Estimating the resistance of an authentication protocol to peeping attacks is very important, but also highly-complex, especially in the case of iChip, as it can simultaneously use many protocol variants, which interfere with each other and further increase their effectiveness (see Appendix). Therefore, we considered them separately, based on the results of related works: [2], [7], [6].

By using the LPN method or its variants, it is possible to efficiently reduce the amount of information leaked about the secret by random injection of erroneous information in the LWE or deterministic rounding in the LWR. Such leakage can also be reduced by modular reduction $x \bmod p$, or by randomly selecting one of several correct LWO options.

It also works similar to introducing an error in the expected, acceptable narrow range. However, introducing noise too much here increases vulnerability to random attacks as well.

As shown in [7]: The k number of arguments used in the function $f(x_1, x_2, \dots, x_k) = x_1 + x_2 + \dots + x_k \bmod p$ depends on the safety function for the statistical algorithm $r(f) = k/2$, however, f cannot be linear, because then the security for Gaussian Elimination is $g(f) = 0$ and the Equation 1 for LWR implementation in iChip takes only 2 arguments ($k = 2$). Therefore, in this case the secret could be recovered even from $O(n)$ challenge-response samples: $m = n^s$, $s = \min(g, r)$.

TABLE I
COMPARISON OF THE MOST IMPORTANT AND OPTIMAL PARAMETERS.

HGPP [Ref.]	k secret objects	n objects pool	Window size	Key size (bits)	Password space	$s(f)$	Guess Rate /round	No of rounds	\approx Time/Auth. (sec)
HB [2]	15	200	200	70	1.5e+22	2	0.5	20	668
APW [5]	16	200	200	79	8.4e+24	1	0.1	6	348
CAS Hi [4]	60	240	20	187	2.4e+57	1	0.5	20	221
CAS Lo [4]	60	80	80	70	8.9e+21	1	0.25	10	122
Foxtail [3]	14	140	30	60	6.5e+18	1	0.5	20	213
CHC [13]	5	112	83	24	1.4e+8	1	0.22	10	93
HCP [7]	50	50	14	164	1.0e+50	1.5	0.1	6	42
grIDSure [19]	6	25	25	28	2.4e+8	1	0.1	6	4
iChip256	36	256	256	512	3.2e+154	2	0.1	6	36
ClickChip	31	289	289	500	1.7e+150	2	\approx 0.01	3	21
FlexiChip	31	256	256	490	1.0e+146	2	0.1	6	15

Fortunately, the introduction of noise by the LWO method in iChip brings the same effect as LPN in the HB [2], which does not allow the simple use of Gaussian Elimination, and the adversary needs to see $O(n^2)$ samples to reveal the secret, also in the case of secret's low entropy [14].

On the other hand, introducing an exception *I gives up to 2 additional arguments by Eq. 2 to this base function, hence $2 \leq k \leq 4$. The number of these arguments is not constant but varies randomly in each challenge, so the Eq. 1 becomes highly nonlinear, especially since V_{out} is the result of a previously used mapping,

Any statistical adversary needs approximately $m = n^{r(f)/2}$ samples to recover the secret, where n is the key size (512 bits for iChip), therefore, if both exceptions (*I and *O) are used then estimated safety function is limited by: $s = \min(g, r) = \min(2, 2) = 2$, hence $m \approx 262,144$ challenge-response samples are needed to reveal the secret.

We tested the resistance of the protocol against finding the secret key with an advanced Genetic Algorithm, which ran for $m=1,000$, $m=10,000$ and $m=20,000$ samples over several days on a computer with an 18-core CPU (Intel i9). The secret created in the default grid size of $N = 256$ but without exceptions (*I, *O) was found after approx. 2 hours of operation. After introducing the LWO, the cracker found a secret key only for microparameters i.e. $N = 25$ (Fig. 3).

With the simultaneous inclusion of *I and *O exceptions, the 2-days search did not give a correct result even for $N = 49$, represented by the microkey in Fig. 9.

&	Inputs	Outputs
1	2 3 4 5 6 7 8 9	1 2 3 4
0		00606606
1	202122120203	13
2	645444454636	35
3	4041425262	3315
4		

Fig. 9. Microkey 7x7 with exceptions *I & *O.

The tests conditions and results are available online [25].

VII. RELATED WORK

Referring to the data in Table 1 of the article from 13th NDSS [12] and the latest publications until today, we have compiled in Table 1, the parameters of the best Human Generated Passwords Protocols, that were created in the years 1991-2017 (there is no significant contribution after 2017), as a comparison with iChip. As we can see, the iChip's parameters have a significant advantage over all others, both in terms of security (key size, keyspace, $s(f)$) and usability (secret's memorizing and authentication time closest to grIDSure). Only HB, HCP, and iChip are protected against linearization [9], where $m = O(n^s)$ strongly depends on the key size n .

The enhanced versions of HB+, Foxtail+ also offer protection against active attacks, but only ClickChip and HHMAC are suitable for the user due to the required authentication time.

VIII. CONCLUSIONS

The result of our work are the iChip protocol and two TurboChip overlays, which significantly accelerate the OTP generation process and all these variants meet the safety and usability criteria required for commercial implementation.

- The FlexiChip is our favorite due to the smaller workload for the user and flexibility in generating passwords of any length L from 1 to $k = |S|$. As the video tutorial [25] shows, the 6-digit OTP computation time easily reaches 15 seconds. The protection against active adversary attacks is provided by the HHMAC with use of standard hash algorithm, preferable SHA-256 or SHA-512. Such signature can be also performed by the user offline on paper documents without any gadgets and automatically scanned and loaded into the system, where this signature is verified.

- The ClickChip overlay has the advantage of using the preliminary round to immediately detect an active adversary attacks and is more glamorous, but requires more proficiency in determining matrix coordinates. At the level of quasi automatic distance evaluation, the authentication time could be reduced to even 15 seconds, similar to the FlexiChip.

The iChip protocol parameters are well-suited to applicate also in resource-constrained devices, like IoT or RFID.

REFERENCES

- [1] T. Matsumoto, H. Imai. *Human Identification Through Insecure Channel*. EUROCRYPT 1991. https://doi.org/10.1007/3-540-46416-6_35
- [2] N. Hopper and M. Blum. *A Secure Human-Computer Authentication Scheme*. Lecture Notes in Computer Science, 2248, 2000.
- [3] S. Li, H.-Y. Shum. *Secure Human-Computer Identification (Interface) Systems against Peeping Attacks: SecHCI*. IACR's Cryptology ePrint Archive: Report 2005/268.
- [4] D. Weinshall. *Cognitive authentication schemes safe against spyware*. IEEE Symposium on Security and Privacy (S&P), 2006.
- [5] H. J. Asghar, J. Pieprzyk, H. Wang. *A New Human Identification Protocol and Coppersmith's Baby-Step Giant-Step Algorithm*. Applied Cryptography and Network Security, 349-366, 2010.
- [6] M. Monteiro, K. Kahatapitiya, H. J. Asghar, K. Thilakarathna, T. Rakotoarivelo, D. Kaafar, S. Li, R. Steinfeld, J. Pieprzyk. *Foxtail+: A Learning with Errors-based Authentication Protocol for Resource-Constrained Devices*. IACR's Cryptology ePrint Archive, Report 2020/261.
- [7] J. Blocki, M. Blum, A. Datta., S. Vempala. *Toward human computable passwords..* ITCS 2017. <https://doi.org/10.4230/LIPIcs.ITCS.2017.10>
- [8] M. Blum, S. Vempala. *Publishable humanly usable secure password creation schemas*. AAAI Conference on Human Computation and Crowdsourcing, HCOMP, 32-41, 2015.
- [9] H. J. Asghar, R. Steinfeld, S. Li, M. A. Kaafar, J. Pieprzyk. *On the Linearization of Human Identification Protocols: Attacks Based on Linear Algebra, Coding Theory, and Lattices*. IEEE Transactions on Information Forensics and Security, 10(8), 1643-1655, 2015.
- [10] S. Samadi, S. Vempala, A. T. Kalai. *Usability of humanly computable passwords*. In arXiv preprint arXiv:1712.03650, 2017.
- [11] A. Juels and S. Weis. *Authenticating Pervasive Devices with Human Protocols*, Advances in Cryptology - CRYPTO 2005, vol 3621.
- [12] Q. Yan, J. Han, Y. Li, R. H. Deng. *On Limitations of Designing Usable Leakage Resilient Password Systems: Attacks, Principles and Usability*. 19th Network and Distributed System Security Symposium (NDSS), 2012.
- [13] S. Wiedenbeck, J. Waters, L. Sobrado, and J.-C. Birget. *Design and evaluation of a shoulder-surfing resistant graphical password scheme*. In Proceedings of the working conference on Advanced visual interfaces, pages 177-184, 2006. <https://doi.org/10.1145/1133265.1133303>
- [14] J. Alwen, S. Krenn, K. Pietrzak, D. Wichs. *Learning with Rounding, Revisited*. Advances in Cryptology - CRYPTO 2013.
- [15] A. Bogdanov, S. Guo, D. Masny, S. Richelson, A. Rosen. *On the Hardness of Learning with Rounding over Small Modulus*, Cryptology ePrint Archive, Report 2015/769.
- [16] I. Dumontheila. *Development of abstract thinking during childhood and adolescence: The role of rostralateral prefrontal cortex*. Developmental Cognitive Neuroscience, 57-76, 2014.
- [17] S. Patil, S. Mercy, N. Ramaiah. *A brief survey on password authentication*. International Journal of Advance Research, Ideas and Innovations in Technology, 4(3), 943-946, 2018.
- [18] F. Wang, L. Leng, A. Teoh, J. Chu. *Palmpoint False Acceptance Attack with a Generative Adversarial Network (GAN)*. Applied Sciences, 10, 8547, 2020. <https://doi.org/10.3390/app10238547>
- [19] S. Brostoff, P. Inglesant, A. Sasse. *Evaluating the usability and security of a graphical one-time PIN system*, Proceedings of the BCS-HCI 2010, Dundee, United Kingdom, 2010.
- [20] R. Jhavar, P. Inglesant, N. Courtois and M. A. Sasse. *Strengthening the security of graphical one-time PIN authentication*. 5th International Conference on Network and System Security, 2011.
- [21] Z. Golebiewski, K. Majcher, F. Zagorski, M. Zawada. *Practical Attacks on HB/HB+ Protocols*. ePrint Archive, Report 2008/241.
- [22] K. Sadeghi, A. Banerjee, J. Sohankar and S. K. S. Gupta. *Geometrical Analysis of Machine Learning Security in Biometric Authentication Systems*, 16th IEEE International Conference on Machine Learning and Applications (ICMLA), 309-314, 2017.
- [23] Y. Sadqi, Y. Belfaïk, S. Safi. *Web OAuth-based SSO Systems Security..* Proceedings of the 3rd International Conference on Networking, Information Systems & Security. NISS 2020.
- [24] A. F. Baig, S. Eskeland. *Security, Privacy, and Usability in Continuous Authentication*. A Survey. Sensors 21, 5967, 2021.
- [25] "Project lab for i-Chip authentication". (July 3, 2022). [Online]: <https://www.researchgate.net/profile/i-Chip-Authentication>

APPENDIX A

INCREASING THE ICHIP ENTROPY

In the sample of 920 students of our university, no secret pattern was repeated. However, the entropy of iChip can be effectively increased by further increasing the key size as shown in subsections A and B, or by using of a suitable background image, which can be built individually by the user or proposed by the wizard as a random structure. The visual structure of such a background image (e.g., Fig. 10) provides reference points for easy remembering the image of the secret and thus allows building a secret key with much higher practical entropy.

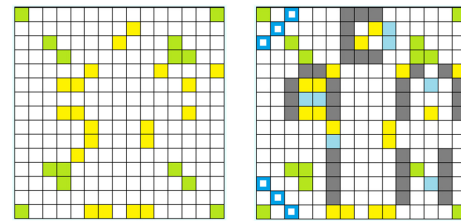


Fig. 10. An example of a secret key depicting the word *admin* on an individually designed background.

A. The iChip as multi-layout interface with 3D key

The iChip interface allows the user to expand the secret not only in 2 dimensions (row x and column y), but also use the 3rd dimension, i.e., the z parameter used here as a layout index (default $z = 1$), marked in colour of the secret elements and is indicated in the i -th round by $z = V_G[i + 1]$. In this simple way, the key size can even exceed a thousand bits, without increasing the C size, where $N_{|z|=1} = x \cdot y = |C| \approx 256$.

Fig. 11 shows an example of a 3D secret that has been adapted from the secret key in Fig. 4, where $|z| = 2$, $N_2 = 450$, by adding 2 blocks on 4 additional layers resulting in $|z| = 6$ and $N_6 = 3 \cdot N_2 = 1350$.

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e
0	1	2	3	4	5	6	7	8	9	04	15	26	28	19	0a
1	4	4	5	4	6	4	4	4	4	33	3	4	4	4	4
2	8	9	4	4	a	5	4	4	4	3b	a	8	4	4	4
3	a	9	a	9	8	a	4	4	4	a8	4	4	4	4	4
4	8	a	9	a	4	a	4	4	4	3b	4	4	4	4	4
5	c	4	c	5	c	6	c	8	9	d7e7	4	4	4	4	4
6	4	7	5	7	7	7	7	9	7	b7c7	4	4	4	4	4
7	7	3	7	2	2	2	2	3	3	74	4	4	4	4	4
8	7	7	c	7	d	6	b	c	4	7a	4	4	4	4	4
9															

2	5	0	1	4	4	2
3	3	0	2	1	7	1
5	7	9	2	3	6	
3	4	2	3	4	8	
9	5	6	3	1	1	8
7	2	1	6	3	1	8
6	8	4	4	2	9	5
4	3	3	2	3	0	4

Fig. 11. An example of a 3D secret on multi-layout iChip.

The zoomed fragment of the challenge image in Fig. 11 shows the first round ($i = 1$) of the OTP calculation: Since $z = V_G[i + 1] = 7$, the search for a value of $V_G[1] = 4$ must start in block 7. It appears in the last element of this block $B_7^I[5]$ at position (6, 2), so we read the value of 3 at the associated output in cell (7, 4) and calculate: $OTP[2] = (V_G[1] + C[(7, 4)]) \bmod 10 = (4 + 3) \bmod 10 = 7$.

B. The iChip as multi-protocol platform

A solution with a large number of protocol variants that can be combined with each other using a simple settings manager is beneficial for increasing its resistance, and makes the scheme more user-friendly, who only needs to know the variants he choose to create his secret. For example, choosing 5 out of 50 variants, the cracker has to check 2,118,760 additional combinations, which must first be analyzed and coded in such a cracker. This number can still expand as each subtle change in protocol represents a new variant that can be created not only by the scientist, but also by the creative user.

The iChip enables the implementation of other Human-Computable Password Protocols, and acts as an open platform for them. We invite other researchers to use it to compose their own licensed variant of HCPP or an adaptation of a previously developed one.

As an example implementation, we used the Foxtail scheme proposed in 2005 by Li and Shum in [3], adapted in 2020 for the needs of IoT in [6]. There are 4 pass-objects in the challenge given in Fig. 12, hence the response $R = 4 \bmod 2 = 0$. For a standard 6-digit OTP, this procedure must be repeated for 20 rounds, each for 1 bit.

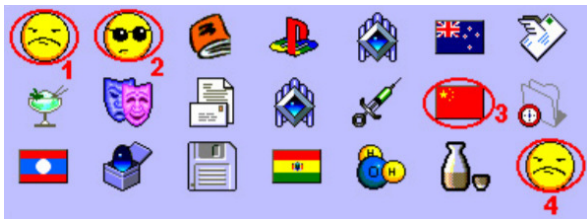


Fig. 12. An example of an original Foxtail challenge for 1 bit response marked by the four red rings.

For implementing the Foxtail protocol on the iChip platform, selected input elements are used as a hidden chain of challenge window in the Foxtail schema. To redirect the binary response of Foxtail, we use two output elements appropriate for the expected binary response (0 or 1) and any 3rd output as an option for the LWO method. Therefore, instead of four protocol rounds for each OTP digit, only one round is needed. To define a trigger for switching between Foxtail and iChip subprotocols, we assume that if the value searched for $V_G[i]$ is found in the first block, then all input elements are treated as a challenge for the Foxtail scheme. Otherwise, for the iChip rules. As counted pass-objects, we assume the value of V_{inp}^i . After adapting the example in Fig. 4 for the Foxtail implementation illustrated in Fig. 13: The first element $V_G[1]$ at position (0, 4) has a value of 8, which appears in the first block; therefore, all input elements in S are treated as a Foxtail challenge. As there are 3 entries with the value of 8, in cells [(6, 4), (5, a), (c, 5)], the response is $3 \bmod 2 = 1$.

However, according to the redirection rules, we use this response for binary addressing of the 2nd element from 2 locations: (d, 7) for 0 and (e, 7) for 1.

In cell (e, 7) is a value of 7, therefore, finally: $OTP[1] = (8 + 7) \bmod 10 = 5$. In the 2nd round the $V_G[2] = 1$, which appear at 2 locations only: [(6, 4), (a, a)]. Since, $2 \bmod 10 = 0$, we go to the cell (d, 7) with a value of 6, and calculate $OTP[2] = (1 + 6) \bmod 10 = 7$. The next values in the generator block no longer appear in the first block, so iChip rules apply to them.

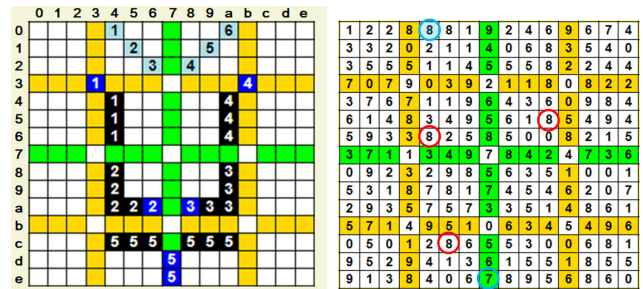


Fig. 13. Image of secret key and Foxtail's pass-objects marked by red rings in the challenge.

APPENDIX B

THE ICHIP IMPLEMENTATION IN THE E-BANKING SYSTEM

We have implemented the iChip protocol in the e-banking system, in which it is used both for logging in and for authorizing transactions. The screenshot in Fig. 14, shows a form for entering the personal data and personalizing the virtual token, visualized as an iPad tablet. The form in the background shows a special wizard for designing of a graphic identifier and secret key, i.e. its image on the right side and the associated code table on the left side. This token implements the iChip protocol and the Turbo-Chip overlays. The token window opens on the form to be authorized. The response for the Click-Chip challenge is shown on the token screen.

The e-banking system is available online [25].



Fig. 14. Screenshot of authorisation by iChip in the e-banking system

Low-complexity access control scheme for MEC-based services

Mariusz Sepczuk, Zbigniew Kotulski, Wojciech Niewolski, and Tomasz W. Nowak
Institute of Telecommunications of WUT, Nowowiejska 15/19, 00-665 Warsaw, Poland
Email: { m.sepczuk, z.kotulski, w.niewolski, t.nowak }@tele.pw.edu.pl

Abstract—The standardized security architecture proposed by ETSI for 5G networks provides six security domains covering network access and secure implementation of network services. However, this architecture does not specify detailed solutions for access control for web services and user credentials management. This paper proposes a new access control and service authorization protocol for the network services using MEC edge servers. Our solution does not slow down the performance of services in the 5G network. The advantage of this solution is that it allows you to solve some network security problems resulting from virtualization techniques (SDN and NFV) applied in constructing contemporary mobile networks.

I. INTRODUCTION

THE FIFTH-GENERATION mobile networks are designed to provide network services with extremely high requirements in terms of bandwidth, the number of devices supported, latency, energy consumption, etc., see [1]. It enables the support of participants of mass events with good quality of service, the implementation of real-time services for telemedicine, monitoring and control of technological processes, and many others. However, it may not be possible to simultaneously meet all 5G quality requirements. Therefore, particular configurations of network services, called slices, have been proposed [2], [3]. In slices, the network parameters are adapted to the specific needs of a given application, for example, a virtual industry instance or its particular use case, see [4]. In order to effectively implement network slices designed for the needs of verticals in the 5G network infrastructure, it is necessary using virtualization techniques. 5G networks use SDN technology [5] to implement network traffic management and NFV technology [6] to enable flexible network deployment and dynamic operation. SDN and NFV technologies also allow the use of modern security solutions integrated with a centralized network controller, for example, the IDS / IPS system, see [7]. Another solution that allows improving network service quality is using edge servers with applications designed to support it. This solution is most effectively implemented with the use of MEC (Multi-access Edge Computing) technology, see [8].

Modern mobile networks of the 5th generation (and higher) are a result of the interaction of SDN and NFV virtualization technologies and MEC edge services technology [9]. From a functional point of view, we can call them 5G MEC networks;

they combine mobile access to resources and services of high quality. Organizationally, such networks require the cooperation of many stakeholders participating in the implementation and use of web services. Ensuring the harmonious collaboration of all participants of the network services market while ensuring the necessary quality requirements and network limitations is a major technological challenge. However, providing the quality of a web service in 5G MEC networks cannot be at the expense of security. In particular, resource and service access protection methods can be a network service bottleneck. Therefore, an access control system adapted to the architecture of web services in 5G MEC networks is necessary for their proper functioning.

This paper aims to propose a new access control scheme for 5G MEC-hosted services that guarantees high service efficiency at a level of security similar to other modern network access control systems. It makes the network providers and the MEC-hosted service providers independent of external identity providers. It ensures strong user authorization to use services in the scope compliant with the applicable security policy. In addition, the system optimizes the operation of the network transmitting data related to the service. It also allows, if necessary, to interact with external identity providers in terms of primary user authentication. This system allows for the improvement of accessing services and granting permissions to users and increases the efficiency of the service implementation process.

The rest of the paper is organized as follows. Section II presents the standardized 5G security architecture and its basic access control domains. Section III reviews modern lightweight network authentication protocols that can be used for authentication and authorization on 5G MEC networks. Section IV presents the outline scheme of the 5G MEC access control architecture used in the special use-cases. Section V defines the optimized access control protocol for 5G MEC-hosted services. Section VI proposes possible improvements and optimizations of components, algorithms, and protocols supporting the access control system. Section VII presents the security advantages of the proposed solution and Section VIII concludes the paper and outlines the future work.

II. 5G NETWORKS AND ACCESS CONTROL

The ETSI standard [10] proposes the 5G network security architecture. The architecture covers essential network security

This paper has been supported by The National Center for Research and Development, Poland, under Decision No. DWM/POLTAJ7/9/2020

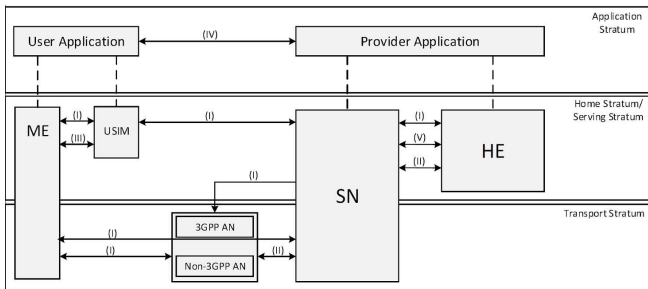


Fig. 1. Overview of the 5G security architecture [10].

elements, particularly network access control and network services access control.

Secure access to 5G MEC web services consists of two steps. The first one is secure 5G network access that enables a UE to authenticate and access the network securely (domain (I) in Fig. 1), including the 3GPP access and Non-3GPP access, and in particular, to protect against attacks on the (radio) interfaces, see [10]. This step is provided by network operators and is based on 5G standardized solutions. The second step is included in the application domain security (marked as domain (IV) in Fig. 1). It consists of security features enabling applications in the user and provider domains to exchange messages securely. This security area is in the competence of the service providers, particularly the MEC services providers. To ensure end-to-end security in 5G networks, the security architecture of the standard [10] should be significantly expanded. We must consider such elements as the end-user devices, the edge server services, and the computing cloud in which the service provider's resources are located in the network. Such an extended security model has been proposed in the paper [11].

The use of MEC technology in 5G networks facilitating the implementation of many services and improving their efficiency requires further expansion of the access control security model. In papers [12] and [13], we proposed a new security architecture with the integrated authorization mechanisms. Before we present our security architecture and the access control security protocols, we will briefly overview the contemporary access control solutions for the web services proposed in the literature.

III. RELATED WORK ON THE ACCESS CONTROL SCHEMES

In mobile networks, the most popular method of access control and authorizing users in web services is applying external identity providers, such as Google, YouTube, Facebook, Okta, Microsoft Active Directory, etc. The offered solutions use well-known authentication and authorization protocols such as OpenId Connect, SAML (Security Assertion Markup Language), and OAuth 2.0. The OpenId Connect [14] is an open standard used for user authentication. It uses JWT (JSON Web Token) to deliver claims about the authentication of an end-user by an authorization server when using a client and other requested claims. SAML [15] plays a similar role but

uses XML to exchange identity credentials and authorization data between identity providers and service providers to verify a user's identity and permissions. OAuth 2.0 provides secure delegated access. The application can take action or access resources from the server on behalf of the user without the user having to share its credentials. It does so by allowing the Identity Provider (IdP) to issue tokens to third-party applications with the user's consent. OAuth 2.0 [16] most often provides authorization services for users who have been authenticated using the OpenId Connect or SAML protocols.

An alternative solution to the authentication and authorization problem may be to use methods that do not require a central identity provider. These include verifiable credentials, decentralized identifiers, and blockchain. Verifiable Credentials (VC) [17] is an open standard for digital credentials. VCs contain such information as context, issuer, type, subject, and identity attributes or a cryptographic proof to ensure their integrity and authenticity. They can be expressed as JWT, see [18]. Decentralized Identifiers (DID) [19] is a type of identifier that enables a verifiable, decentralized digital identity. DIDs contain cryptographic material, verification methods, or service endpoints, which provide a set of mechanisms enabling independent controllers to prove to check the self-sovereign identity, see [20], [21]. Blockchain technology can be an alternative to decentralized authentication and authorization, see, e.g., [22].

In addition to authentication and authorization solutions for services in 5G networks, seen as mobile networks with general architecture, the literature offers proposals for solutions that consider the specific role of edge servers and MEC technology. It is proposed to provide access to enabler systems [23] or proxy servers [24] for user authentication and authorization. In order to improve data transmission, increase its security and improve efficiency, it is proposed to authorize the transmitted packets [25] additionally. Solutions of this type can operate independently or cooperate in identity delivery systems implemented by global suppliers, creating designs with high security, high flexibility, and the necessary autonomy in privacy.

In the literature, you can find works presenting access control protocols prepared especially for the 5G MEC network and services hosted in edge servers. For instance, paper [26] introduced a new access control protocol for the MEC system model and examined this protocol against user privacy requirements when sharing information that is not essential to the edge server-hosted service.

IV. 5G MEC ACCESS CONTROL ARCHITECTURE

Control of access to resources and services is an essential element of computer network security, and in mobile networks, it constitutes the basis for their protection. The access control system is the central element in our proposed 5G MEC network security architecture [12]. This section will outline this architecture, with its security domains and the core component specification. Our security architecture model consists of ten security domains that coexist and cooperate,

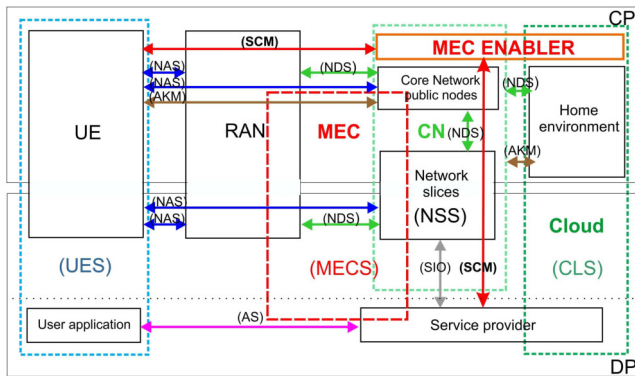


Fig. 2. The high-level access control security architecture in 5G MEC.

providing security services in all aspects of the 5G MEC mobile network functioning. It is related to basic the 5G security architecture presented in Fig. 1 and its extension proposed in [11], however, it contains components required by the MEC technology solutions. It is able to satisfy mobile networks security requirements enabling simultaneously to provide web services satisfying 5G networks' high quality expectations.

The 5G MEC security architecture designed in [12] consists of ten security domains. The number of the domains we consider is higher than those in [10], where six security domains for the 5G network are defined. This is because it must connect four network environments: 5G Core Network (CN), 5G Radio Access Network (RAN), the edge services provided by the MEC technology, and a new module which is the MEC Enabler, see Fig. 2. The domains of responsibility for security in the new architecture partially overlap with the areas proposed in [10]. To a large extent, they implement new security functions resulting from applications of the MEC technology and increase the number of stakeholders in the implementation of mobile network services. Below we present the security domains and their areas of responsibility, see Fig. 2.

- **NAS (Network Access Security)** provides basic security of user data. It includes confidentiality and integrity of signaling and user data between the User Equipment (UE) and the network in the Control Plane (CP) or the Data Plane (DP). It corresponds to the security domain (I) in Fig. 1.
- **NDS (Network Domain Security)** is the secure exchange of signaling and user data between different network entities. It corresponds to the security domain (II) in Fig. 1.
- **UES (User Equipment Security)**: this domain contains software and hardware security at the user's side, including user access to the mobile equipment. It corresponds to the security domain (III) in Fig. 1.
- **AS (Applications Security)** is the support for secure communications between applications in UE and applications offered by the Service Provider. It is under the

control of the end-user or the Service Provider (in the Application sub-Plane).

- **AKM (Initial Authentication and Key Management)**: the security features that enable network functions to communicate securely. It includes mechanisms for authentication and key management that implement the unified authentication framework.
- **SCM (Security Credentials Management)** includes the service authentication and relevant key management between UE and the external data-transmitting network. In our model, the MEC Enabler governs this security domain.
- **SIO (Security Interoperability)** (also via MEC) is the support of the openness of security capability between the 5G network entity and the external Service Provider. It also includes the set of features that enable the stakeholders to know whether a security feature is in operation or not, which is defined as the security domain (VI) in [10].
- **NSS (Network Slices Security)** includes security of slices in terms like access control, authorization, and isolation.
- **MECS (MEC Security)**: protection of Service Provider's software, virtualization platform (VM), and hardware supporting the edge server and the MEC host.
- **CLS (Cloud Security)** includes all solutions to secure resources and communication inside the cloud and the operator's home domain. This domain extends the resources offered by MEC-hosted services to the external cloud resources.

Two other security domains in Fig. 1 are the Application domain security (IV), providing the security features that enable applications in the user domain and the provider domain to exchange messages securely, and the SBA (Service Based Architecture) domain security (V), providing the security features that enable network functions of the SBA architecture to securely communicate within the serving network domain and with other network domains. In our model, their responsibility is distributed over AS, SCM, SIO, NSS, and MECS security domains because of the new important component, the MEC Enabler. The MEC Enabler acts as an AAA (Authentication, Authorization, and Accounting) server for all requests before any connection with MEC. Then, if the request is authorized, the MEC Enabler will properly control the configuration process of access to the MEC infrastructure and will create an information token that will enable to use of selected MEC services, see Fig. 3. The procedure of token creation and protection is an example of the solution analyzed and provided by ETSI (European Telecommunications Standards Institute) [10]. It involves the JSON (JavaScript Object Notation) Web Tokens with the appropriate digital signature, see the series of the RFC documents: [28], [29], [30], [31], [32]. This solution will reduce the management process and network resources' utilization for non-legitimate connections. The MEC Enabler will be responsible for service authentication and relevant

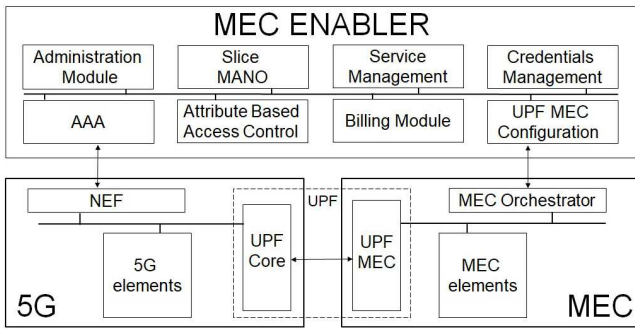


Fig. 3. The new network architecture with the MEC Enabler.

key management between UE and the external data network realizing:

- services for legitimate users of the network,
- services for the secured communication (via slices or protected links),
- main security service for access to the MEC-hosted services,
- management of access rights by credentials,
- giving credentials to the progression of services considered as 5G MEC use cases,
- enabling services in the external operator's domain or over the cloud.
- **Administration Module:** this module supports all management operations and can configure other modules.
- **Slice MANO (Management And Network Orchestration):** the role of this module is to manage the slice life cycle in the MEC area. This manager can also reserve resources for specific slices and mark them according to service type, network identification, or any other rule, [34].
- **Attribute-Based Access Control:** this module stores all policy data about services, slices, available connections, and some other MEC information [35].
- **Service Management:** this module allows checking available services, their utilization, and the number of resources dedicated to them. The specific use cases can also manage the MEC's services lifecycle using MEC orchestrator API. However, it is also necessary to implement load balancing, which analyzes service placement [36].
- **Credentials Management:** this module creates tokens used for authorization of the MEC resources. This module's created token is monitored, refreshed, or deleted according to service needs [37].
- **Billing Module:** this module is dedicated to billing and storing information about MEC usage in different business models [38].
- **UPF MEC Configuration:** this module matches and adequately configures UPF MEC to link proper network slices with dedicated MEC resources [39]. When each slice represents another operator, this module establishes a connection between the operator and the MEC compu-

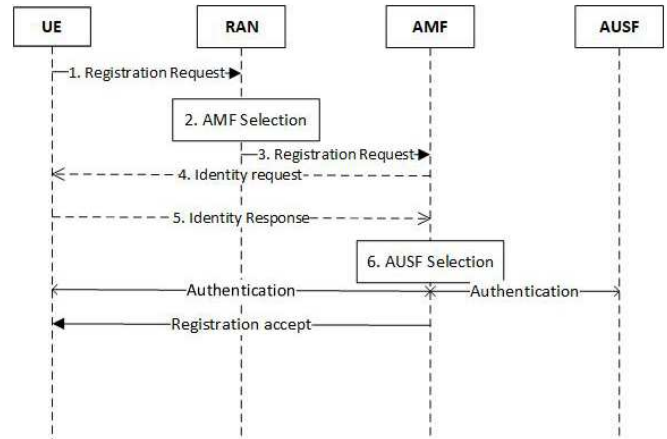


Fig. 4. Process of UE registration to the 5G network

tation part assigned to it.

- **AAA:** this module is responsible for authentication, authorization, and accounting of all requests to MEC services [40]. After positive verification, UPF MEC Configuration Module prepares a network configuration that allows creating a connection with the chosen service, and Credentials Management generates a token for service authorization. Implementation of this module in the MEC Enabler will significantly improve the protection of Edge resources.

V. 5G MEC ACCESS CONTROL PROCEDURE

A. General access control to 5G network

The access control process includes three phases: UE registration to the 5G network, the discovery of proper UPF Core, and access to a MEC service. In the registration part, devices are verified whether they can be connected to the 5G network. In the discovery phase, the network must find a proper UPF that can communicate with the MEC network. Moreover, in this phase possibility of establishing a connection between UPF Core and UPF MEC is checked. Finally, access to selected MEC services is performed when the UPF Core can securely communicate with UPF MEC.

The diagram in Fig. 4 represents the communication flow in phase 1 when the UE tries to register to the 5G network. The flow in the registration phase was created based on the registration procedure shown in ETSI Technical Specification [49]. First of all, UE sends a registration request to the gNB (RAN) with all necessary information about UE (UE context), such as SUCI (Subscription Concealed Identifier), last visited TAI (Tracking Area Identity), Requested NSSA (Network Slice Selection Assistance Information) and many more. Then, the RAN chooses AMF (Access and Mobility Management Function). The AMF performs most of the MME's functions in the 4G network. One of these is UE authentication. AMF sends an identity request to the UE (optional request for additional data), and when it gets a response, AUSF (Authentication Server Function) performs the selection procedure. After that,

authentication messages are exchanged between the AUSF, AMF, and the UE. Finally, if everything goes correctly, the UE is registered as reported by the AMF.

In the phase of discovery proper UPF Core (see Fig. 5), authentication between UE and AUSF is first performed. After that, the AMF module sends a request to SMF (Session Management Function) to start the UPF Core discovery and a selection procedure. In the classical approach to the 5G MEC, network SMF is responsible for UPF management (e.g., configure traffic steering rules), so it must communicate with a new authentication and authorization service to ensure proper access control to MEC. To check if UE can access the MEC service, SMF, through NEF, sends a verification request to MEC Enabler (MEC EBR). The MEC EBR checks if UE via UPF CORE can establish a session with UPF MEC and if UE can use a particular MEC service. As a result of verification, a token is created. The token contains the necessary information as a result of two previous checks. If everything is going well and SMF receives the token, it sends the request to UPF CORE for an update session. Finally, after receiving a response about the correct data update from UPF CORE, a sequence of messages confirming the possibility of sending data from the UE to UPF CORE and UPF MEC is sent.

The final phase of access control to MEC service is internal access in MEC infrastructure (see Fig. 6). UPF MEC sends an application request to TMS (Traffic Management Service). The TMS is responsible for sending service access information to the MEC platforms. The MEC platform chooses adequate service and verifies the received token. After that, the MEC platform sends a request to the selected service; data from the service is sent to the UPF MEC and finally to the UE. In this phase, the proposed message flow is similar to the standard proposed in RFC 7519 for JWT tokens [32], making it easy to implement and use.

The descriptions above relate to the general assumptions made for controlling access to services in the 5G network. In the context of the 5G MEC network, the description should be detailed, and this will be done in the next subsection.

B. Steps of created access control protocol

Before accessing the resources hosted in the MEC environment, some steps should be taken (see Fig. 7). The proposed authorization process includes actions related to registration in the 5G network, which are the first condition for accessing the network [Step 1]. If the first access condition is met, the client will be correctly registered in the network, and then it will be able to try to connect to MEC resources. For this purpose, its request will be authenticated through the second step of the access control process, which verifies whether the request can be authorized by the access server dedicated to the application [Step 3]. After successfully passing the second step, the request is sent for the policy-based access verification [Step 5]. At this point, the access policy is checked based on the knowledge about the device's origin in the network and confirmation of the rights to use the indicated resource. Before making a decision, the policy-based access module

considers many aspects, such as information about the slice from which the request is sent, a destination of the request, the user name, and more. Then, according to the stored policy, MEC Enabler decides whether to send the request to the application or not [Steps 6-7]. Suppose the request complies with the policy and in that case, the network element will be appropriately configured to enable connection from the device to the application located in the MEC [Step 8]. The exact course of the authorization process is presented in the points below:

- 1) The registration request is sent from the User Equipment (UE) RAN Access components:
 - The request is sent to the Access and Mobility Management Function (AMF)
 - Start of the Primary Authentication between UE and Authentication Server Function (AUSF) according to the 5G procedure
 - UE establishes NAS security context with AMF
 - UE initiates the establishment of a new PDU Session by sending NAS message
 - AMF selects V-SMF (Visited Session Management Function) and sends PDU Session context request
 - V-SMF sends PDU Session context response.
 - H-SMF (Home Session Management Function) obtains subscription information from UDM and verifies that UE's request is compliant
 - H-SMF sends EAP Request/Identity message to UE
 - UE sends EAP Response/Identity message
 - H-SMF selects UPF and establishes an N4 Session.
- 2) H-SMF forwards request to UPF
- 3) UPF forwards the request containing EAP Response/Identity message to the MEC Enabler AAA server (ME:AAA).
- 4) AAA module verifies if the sent data are correct. After positive verification, authorized data (user's contextual data and positive authorization) are sent to Slice MANO.
- 5) Slice MANO based on available resources information on local MANO slice resource database extends authorization data with the slice data such as priority of the request, identifiers of requested slice/slices, service localization
- 6) ABAC module checks data collected from Slice MANO in the context of one of the access policies located in a local ABAC policy database. Finally, slice data and ABAC verification information are forwarded to the Credentials Management module after positive verification authorization data.
- 7) Credentials Management entity creates JMAT token based on gathered data from ABAC entity. It sends it with configuration data (service localization, priority of handling requests, required types of UPF, slice ID, and network token) to the UPF Configuration module.
- 8) ME: UPF Configuration module based on configuration data from Credentials Management module sets network configurations on UPF Core and UPF MEC to establish

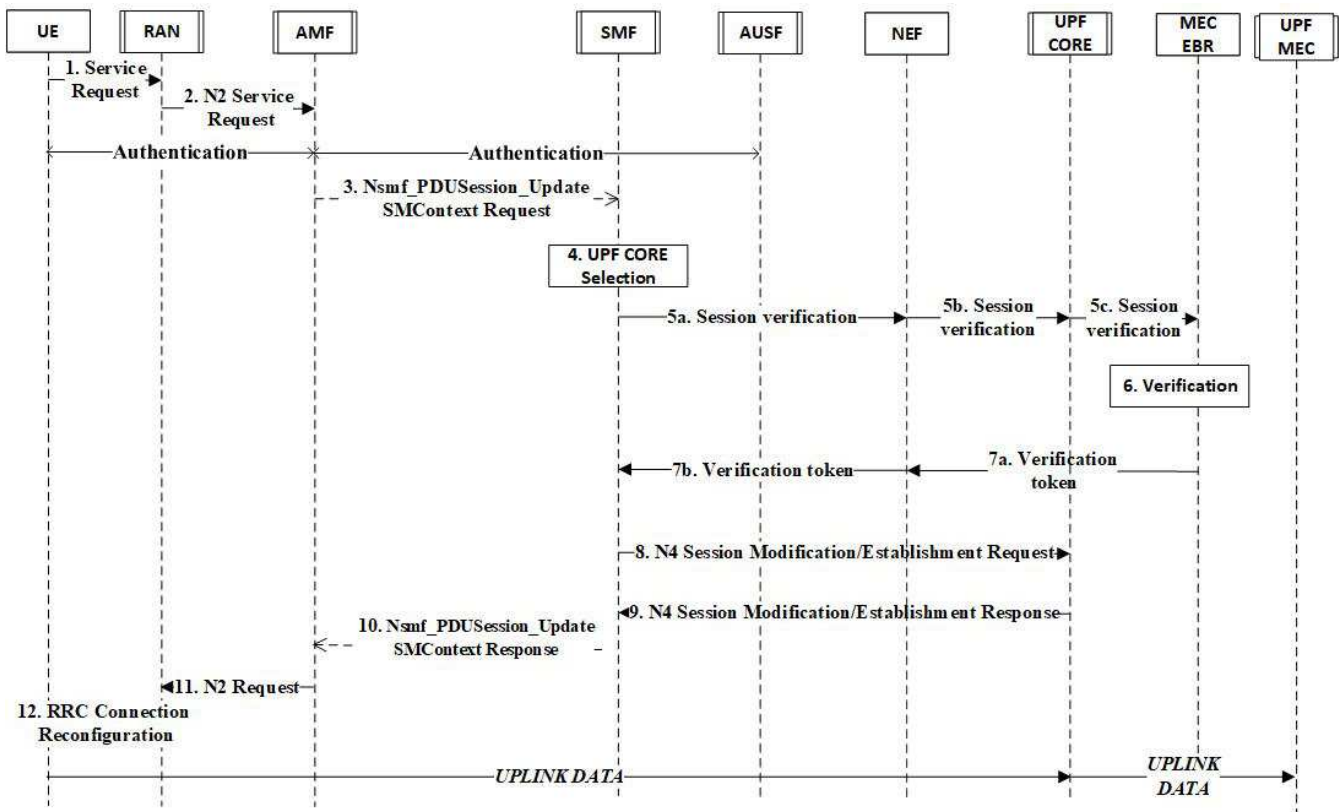


Fig. 5. Process of discovery proper UPF Core

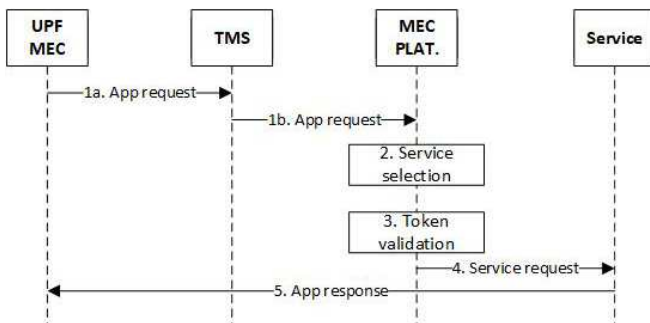


Fig. 6. General process of access to a MEC service

a connection to the service.

- 9) The request is transferred to the service located in the MEC environment.
- 10) UE establishes an End-to-End connection with the requested MEC:App.

As presented in the process description, the central role of the MEC Enabler in the authorization process is to add another access control step (based on verification of additional context information). This step increases the security of connection with the MEC environment and allows the protocol to be extended by other steps in the future. Currently, the policy verification process makes binary decisions - it may allow

access or decline the connection.

Naturally, the data flow from Fig 7 shows the connection part to the MEC service, not the management of the established connection. In addition, the service access token must be verified for its validity (e.g., token lifetime, data validity in the token, etc.).

VI. POSSIBLE IMPROVEMENTS AND OPTIMIZATIONS

This section describes entities that might be improved in the overall architecture presented in this paper.

A. JMAT token generation

In the past, during the design and implementation of JMAT tokens, we decided to improve its structure. The detailed results were described in [13]. Below are the key improvements that reduced the generation time:

- Reorganize the way of storing the data - we utilized a file system with directories and proper file naming as a data structure that exposes the quick find operation.
- Avoid JSON (JavaScript Object Notation) deserialization, also called JSON parsing - deserialization is a widely used operation in Object Oriented Programming, however it might be time-consuming. We decided to utilize the knowledge about the exact object structure that must be deserialized to read needed data.

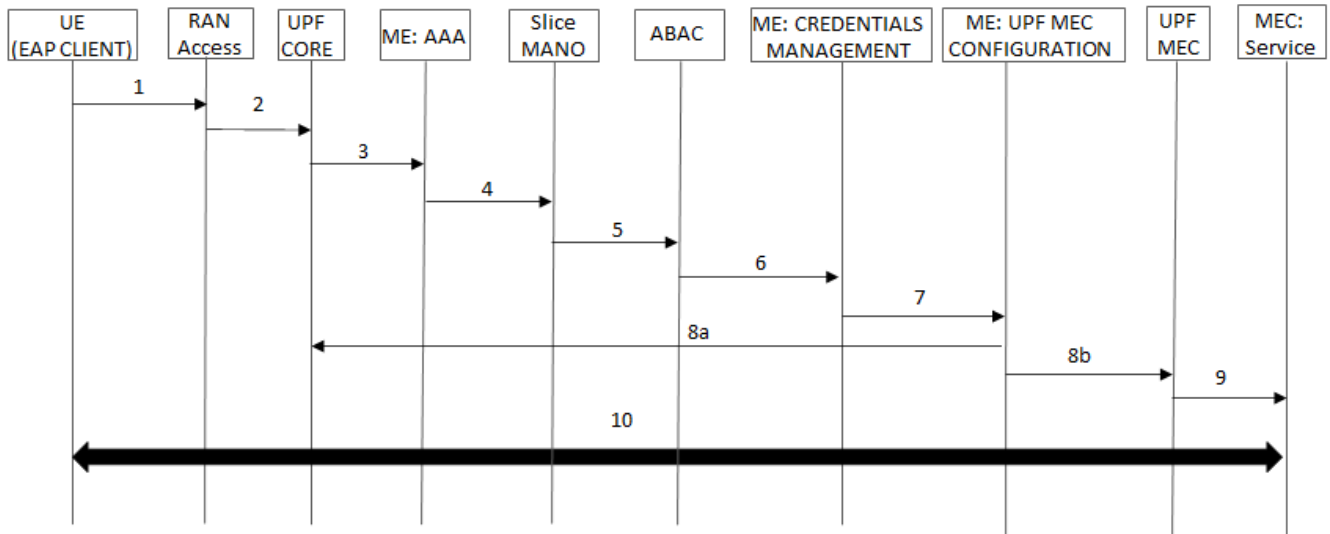


Fig. 7. Steps of created access control protocol

B. Py-ABAC rules and rule evaluation

We utilize ABAC (Attribute-Based Access Control) approach to manage access control to services for users. We use the py-ABAC library, described widely in [27]. Incoming requests might be denied or allowed with py-ABAC policies evaluation during Step 6 in the process in Fig. 5. The py-ABAC engine evaluates the final boolean decision (deny or allow the request) with rules stored in the memory or database. Each rule gives a boolean result, and the final result is calculated by n -ary OR or AND operation over those rules. In the worst scenario, the engine requires all rules to be evaluated to obtain the result. This algorithm could be improved by properly ordering the rules first to check those rules that determine the final decision. As a result, the average time needed for request analysis might decrease if rules are ordered properly. The first rule to evaluate might also be selected by a classification algorithm executed with the request's data and context.

C. Token verification after authorization

According to the concept of MEC Enabler, each first service request must be checked if it is authorized to the service and if it complies with the access policy. After passing the authorization process, an access token is generated. This token contains important information that confirms connection access to the service and data for authentication.

When the service implementation status does not change, the token validation causes unnecessary delays because the other parameters are the same apart from the token validity. Limiting the validation process only to check the token's validity would be sufficient.

When a service implementation status has changed, the token validation should be extended with additional policy verification. It would be sufficient first to limit verification with altered parameters and check if they fulfill policy needs.

The expected benefit is to reduce the time and resources needed for token validation. This approach might expose the authentication engine to some attacks based on token spoofing attempts or cheating the token policy.

D. Early evaluation

Incoming requests require authentication. The system could classify the request as pre-approved or pre-rejected based on the request's header and content (the early evaluation). When the request is pre-approved, the needed resources to establish the connection are collected, and the link is created before the request is authenticated and authorized. The expected benefit is the reduction of the latency for link creation. The main risk is the unavailability of needed resources for other valid requests due to pre-approvals. The classification algorithm might be applied here. Every decision made by the algorithm is validated by the authentication and authorization process, so with every request, the accuracy of the classification might be better. The over-learning problem should be addressed, e.g., by some data retention.

The approved request could be handled by MEC service or external (cloud) service. Thus, the early evaluation might return the following results: deny the request, take by MEC service, handle by cloud service.

VII. SECURITY ASPECTS OF THE PROPOSED SOLUTION

As we presented in the Introduction, the 5G mobile networks take full advantage of virtualization technologies (SDN and NFV) and the location of services in edge servers (MEC) to guarantee the highest quality of services. However, all these technologies require an innovative approach to the problem of network security and the security of MEC applications. On the one hand, virtualization technologies and shared infrastructure increase the network's vulnerability to attacks due to the openness of components that perform data transmission and other

network functionalities. Another issue is the fact that different stakeholders simultaneously use network functionalities. On the other hand, the transparent structure of the network with centralized traffic control allows the creation new level of security supervision. For example, a network controller with SDN technology can validate packages for different aspects. Consequently, it can be used to build a uniform decentralized IDS protection system covering all network nodes and supervising the correctness of packet flow, see [43]. In this case, responsibility for the entire system relies on the controller side. The problem of the effect of using programmable network technology on the network's security is extensively studied and has extensive literature, see, e.g., [44].

Among the various security methods proposed in the modern 5G MEC networks, an independent access control and user rights management system can significantly improve network security. The new access control architecture proposed by us, with the central element of the MEC Enabler, fully meets this expectation. It is compatible with network components belonging to different operators. Thus, in the control plane of SDN-based networks, a crucial vulnerability is the centralized single point control resulting in a global view of the network and exposing the underneath topology of a system [41]. In the literature, the proposed remedy is extending the Open Flow protocol to enable communication of the security policies between the security applications in the Controller to the agents in the switches, see [45]. In our solution, the MEC Enabler can improve the security of connections to the MEC by an additional layer of traffic control. Even when the MEC Enabler system is down, this will not impact the connectivity because the network without additional protection will be managed in a legacy manner (of course, in this case, security improvement done by MEC Enabler will not be available).

In the data plane, a critical vulnerability is that there is no standardized authentication mechanism in the switch for input traffic or incoming buffer data. Thus, erroneous flow alternation is possible. In the literature, the possible solutions could be using a covert channel defender (CCD), which can efficiently detect and prevent rule conflicts in the data plane, see [46]. It can also be an application of the access control scheme dedicated to a network distributed Intrusion Prevention Systems [47]. In our security architecture, the MEC Enabler can authenticate traffic based on JMAT tokens and, what is more, filter all traffic that has no valid token. Authentication done by MEC Enabler is multidimensional and includes more network information about connection such as slice, network provider, requested MEC application, and others.

Finally, a dangerous vulnerability is that there is no mechanism for identity control in the end-host/control channel. In the literature, the proposed solution can be cryptographic unique message identification for each LLDP packet, see [48]. The MEC Enabler can authenticate and check privileges for all MEC connections by analyzing the JMAT token. This type of access policy implemented in the MEC Enabler is essential for all edge systems. It protects its limited resources from unnecessary consumption and against dangerous attacks that

automatically drop before reaching the MEC environment.

VIII. CONCLUSIONS AND FUTURE WORK

In the paper, new access control and service authorization protocol for the network services using MEC edge servers was described. Firstly, we presented the standardized 5G security architecture and its essential access control domains. After that, the reviews of modern lightweight network authentication protocols that can be used for authentication and authorization on 5G MEC networks were specified. Next, we described the newly created access control procedure: starting with the characterization of MEC Enabler as the main element of the proposed solution, through the interaction of all 5G MEC network elements in the implementation of access, and finally presenting the advantages of the solution in terms of security.

In future works, we will focus on testing the created access control process for selected services in our experimental environment. We will extend the test bed with the implementation of new 5G MEC architecture components and prepare their verification in terms of the requirements enforced on 5G network services. Moreover, we plan to optimize the access control system according to several criteria, e.g., the applied resources and operations costs, its impact on QoS/QoE, expected risks and security level (Quality of Protection), etc. Finally, we would like to improve authentication mechanisms in the MEC Enabler by using Machine Learning algorithms for advanced policy analysis to reduce the time needed for the authentication and authorization procedure.

REFERENCES

- [1] *Minimum requirements related to technical performance for IMT-2020 radio interface(s)*. Report ITU-R M.2410-0, ITU (2017)
- [2] *Dynamic end-to-end network slicing for 5G*, Nokia White Paper (2016). Global mobile Suppliers Association. <https://gsacom.com>
- [3] Z. Kotulski, T. Nowak, M. Sepczuk, M. Tunia, R. Artych, K. Bocianiak, T. Osko, and J.-P. Wary, "On end-to-end approach for slice isolation in 5G networks. Fundamental challenges," *2017 Federated Conference on Computer Science and Information Systems (FedCSIS)*, 2017, pp. 783-792. <https://doi.org/10.15439/2017F228>
- [4] T.W. Nowak, M. Sepczuk, Z. Kotulski, W. Niewolski, R. Artych, K. Bocianiak, T. Osko, and J.-P. Wary, "Verticals in 5G MEC-Use Cases and Security Challenges," *IEEE Access*, vol. 9, pp. 87251-87298, 2021, <https://doi.org/10.1109/ACCESS.2021.3088374>
- [5] A.A. Barakabitze, A. Ahmad, R. Mijumbi, and A. Hines, "5G network slicing using SDN and NFV: A survey of taxonomy, architectures and future challenges," *Computer Networks*, vol. 167, 11 February 2020, 106984. <https://doi.org/10.1016/j.comnet.2019.106984>
- [6] ETSI GS NFV-IFA 010: *Network Functions Virtualisation (NFV) Release 3; Management and Orchestration; Functional requirements specification, V3.6.1*, ETSI (2022-01).
- [7] F.N. Nife and Z. Kotulski, "Application-aware firewall mechanism for Software Defined Networks," *J Network and System Management* **2020**, vol. 28, pp. 605-626. <https://doi.org/10.1007/s10922-020-09518-z>
- [8] Y. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, *Mobile Edge Computing. A key technology towards 5G*. **2015**, ETSI White Paper No. 11.
- [9] B. Blanco et al., "Technology pillars in the architecture of future 5G mobile networks: NFV, MEC and SDN," *Comput. Stand. Interfaces*, **2017**, vol. 54(4), pp. 216-228, <https://doi.org/10.1016/j.csi.2016.12.007>
- [10] *5G; Security architecture and procedures for 5G System*. ETSI TS 133 501 V16.5.0 (2021)
- [11] X. Ji, K. Huang, L. Jin, H. Tang, C. Liu, Z. Zhong, W. You, X. Xu, H. Zhao, J. Wu, and M. Yi, "Overview of 5G security technology," *SCIENCE CHINA, Information Sciences*, **61** 081301:1-081301:25 (2018) <https://doi.org/10.1007/s11432-017-9426-4>

- [12] Z. Kotulski, W. Niewolski, T. Nowak, M. Sepczuk, "New Security Architecture of Access Control in 5G MEC," in: *Thampi, S.M., Wang, G., Rawat, D.B., Ko, R., Fan, C.I. (eds) Security in Computing and Communications. SSCC 2020*. Communications in Computer and Information Science, vol. 1364. Springer, Singapore 2021. https://doi.org/10.1007/978-981-16-0422-5_6
- [13] W. Niewolski, T.W. Nowak, M. Sepczuk, Z. Kotulski, "Token-based authentication framework for 5G MEC mobile networks," *Electronics*, 2021, vol. 10, 1724. <https://doi.org/10.3390/electronics10141724>
- [14] *Welcome to OpenID Connect*, <https://openid.net/connect/>
- [15] *Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0 - Errata Composite*, Working Draft 07, 8 September 2015, <https://www.oasis-open.org/committees/download.php/56782/sstc-saml-profiles-errata-2.0-wd-07.pdf>
- [16] D. Hardt, Ed., *The OAuth 2.0 Authorization Framework*, RFC 6749, October 2012, Available online: <https://datatracker.ietf.org/doc/html/rfc6749>
- [17] *Verifiable Credentials Data Model v1.1*, W3C Recommendation 03 March 2022 <https://www.w3.org/TR/vc-data-model/>
- [18] N. Fotiou, V.A. Siris, and G.C. Polyzos, "Capability-based access control for multi-tenant systems using OAuth 2.0 and Verifiable Credentials," *arXiv:2104.11515v2 [cs.CR]* 28 Apr 2021 <https://arxiv.org/abs/2104.11515>
- [19] *Decentralized Identifiers (DIDs) v1.0. Core architecture, data model, and representations*, W3C Proposed Recommendation 03 August 2021 <https://www.w3.org/TR/did-core/>
- [20] A. Preukschat, D. Reed *Self-sovereign identity: decentralized digital identity and verifiable credentials*, Manning (June 8, 2021), ISBN-13: 978-1617296598.
- [21] J. Sedlmeir, R. Smethurst, A. Rieger, and G. Fridgen, "Digital Identities and Verifiable Credentials", *Bus Inf Syst Eng*, vol. 63(5), pp. 603-613, 2021. <https://doi.org/10.1007/s12599-021-00722-y>
- [22] N. Fotiou, I. Pittaras, V.A. Siris, S. Voulgaris, and G.C. Polyzos, "OAuth 2.0 authorization using blockchain-based tokens," *arXiv:2001.10461v1 [cs.CR]* 28 Jan 2020 <https://arxiv.org/abs/2001.10461>
- [23] B. Liang, M.A. Gregory, S. Li, "Multi-access Edge Computing fundamentals, services, enablers and challenges: A complete survey," *Journal of Network and Computer Applications* vol. 199 (2022) 103308. <https://doi.org/10.1016/j.jnca.2021.103308>
- [24] A. Ali, S.R. Khan, S. Sakib, S. Hossain, and Y.-D. Lin, "Federated 3GPP Mobile Edge Computing systems: a transparent proxy for third party authentication with application mobility support," *IEEE Access*, vol. 10, pp. 35106-35119, 2022. <https://doi.org/10.1109/ACCESS.2022.3162851>
- [25] Sakthibalan Pandiyan, Devarajan Krishnamoorthy, "NRTAS: Non-redundant traffic authentication scheme for strengthening privacy in 5 G communication networks," *Journal of Intelligent and Fuzzy Systems*, April 2022. <https://doi.org/10.3233/JIFS-212750>
- [26] G. Akman, P. Ginzboorg, and V. Niemi, "Privacy-Aware Access Protocols for MEC Applications in 5G," *Network* 2022, 2, pp. 203-224. <https://doi.org/10.3390/network2020014>
- [27] *Project description: py-ABAC. Attribute Based Access Control (ABAC) for python*. <https://pypi.org/project/py-abac/0.2.0/>
- [28] *JSON Web Signature (JWS)*, RFC 7515 (2015) Available online: <https://tools.ietf.org/html/rfc7515>
- [29] *JSON Web Encryption (JWE)*, RFC 7516 (2015) Available online: <https://tools.ietf.org/html/rfc7516>
- [30] *JSON Web Key (JWK)*, RFC 7517 (2015) Available online: <https://tools.ietf.org/html/rfc7517>
- [31] *JSON Web Algorithms (JWA)*, RFC 7518 (2015) Available online: <https://tools.ietf.org/html/rfc7518>
- [32] *JSON Web Token (JWT)*, RFC 7519 (2015) Available online: <https://tools.ietf.org/html/rfc7519>
- [33] *Functional architecture and information flows to support Common API Framework for 3GPP Northbound APIs*. 3GPP TS 23.222 V17.4.0 (2021-04)
- [34] Z. Kotulski, T. Nowak, M. Sepczuk, M. Tunia, R. Artych, K. Bocianiak, T. Osko, and J.-P. Wary, "Towards constructive approach to end-to-end slice isolation in 5G networks," *EURASIP J. Information Security* 2018, 2 (2018). <https://doi.org/10.1186/s13635-018-0072-0>
- [35] W. Fisher, N. Brickman, P. Burdenet et al., *Attribute Based Access Control*. NIST SP 1800-3, Second draft (2017)
- [36] B. Brik, P.A. Frangoudis, A. Ksentini, "Service-oriented MEC applications placement in a Federated Edge Cloud Architecture," in: *IEEE Int. Conf. on Communications (ICC), Dublin, Ireland, 2020*, pp. 1-6. <https://doi.org/10.1109/ICC40277.2020.9148814>
- [37] P.A. Grassi, M.E. Garcia, J.L. Fenton, *Digital Identity Guidelines*. NIST SP 800-63-3 (2017). <https://doi.org/10.6028/NIST.SP.800-63-3>
- [38] *Multi-access Edge Computing (MEC): Phase 2: Use Cases and Requirements*. ETSI GS MEC 002 V2.1.1 (2018-10)
- [39] *Multi-access Edge Computing (MEC). MEC 5G Integration*. ETSI GR MEC 031 V2.1.1 (2020-10)
- [40] S. Behrad, E. Bertin, N. Crespi, "A survey on authentication and access control for mobile networks: from 4G to 5G," *Ann. Telecommun.* **2019**, vol. 74, pp. 593-603. <https://doi.org/10.1007/s12243-019-00721-x>
- [41] R. Deb and S. Roy, "A comprehensive survey of vulnerability and information security in SDN," *Computer Networks*, Volume 206, 7 April 2022, 108802, <https://doi.org/10.1016/j.comnet.2022.108802>
- [42] *NFV Security in 5G - Challenges and Best Practices*, ENISA Report, February 24, 2022, <https://www.enisa.europa.eu/publications/nfv-security-in-5g-challenges-and-best-practices> <https://doi.org/10.2824/166009>
- [43] F. Nife, Z. Kotulski, and O. Reyad, "New SDN-oriented distributed network security system," *Appl. Math. Inf. Sci.* vol. 12, no. 4, pp. 673-683 (2018) <https://doi.org/10.18576/amis/120401>
- [44] Y. Maleh, Y. Qasmaoui, K. El Gholami, Y. Sadqi, and S. Mounir, "A comprehensive survey on SDN security: threats, mitigations, and future directions," *Journal of Reliable Intelligent Environments*, 2022. <https://doi.org/10.1007/s40860-022-00171-8>
- [45] K.K. Karmakar, V. Varadharajan, and U. Tupakula, "On the design and implementation of a security architecture for Software Defined Networks," in *2016 IEEE 18th International Conference on High Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, 2016, pp. 671-678, <https://doi.org/10.1109/HPCC-SmartCity-DSS.2016.0099>
- [46] Q. Li, Y. Chen, P.P.C. Lee, M. Xu, and K. Ren, "Security Policy Violations in SDN Data Plane," *IEEE/ACM Transactions on Networking*, vol. 26, no. 4, pp. 1715-1727, Aug. 2018, <https://doi.org/10.1109/TNET.2018.2853593>
- [47] F. Nife, Z. Kotulski, "New SDN-Oriented Authentication and Access Control Mechanism," in: *Gaj, P., Sawicki, M., Suchacka, G., Kwiecien, A. (eds) Computer Networks. CN 2018. Communications in Computer and Information Science*, vol 860. Springer, Cham 2018. https://doi.org/10.1007/978-3-319-92459-5_7
- [48] T. Alharbi, M. Portmann, and F. Pakzad, "The (in)security of topology discovery in OpenFlow-based software defined network," *Int. J. Netw. Secur. Appl.* 10 (2018) 01-16. <https://doi.org/10.1109/LCN.2015.7366363>
- [49] ETSI Technical Specification, *5G; Procedures for the 5G System (5GS) (3GPP TS 23.502 version 16.5.0 Release 16)* **2022**, https://www.etsi.org/deliver/etsi_ts/123500_123599/123502/16.05_00_60/ts_123502v160500p.pdf

Advances in Information Systems and Technologies

AIST is a FedCSIS conference track aiming at integrating and creating synergy between disciplines of information technology, information systems, and social sciences. The track addresses the issues relevant to information technology and necessary for practical, everyday needs of business, other organizations and society at large. This track takes a socio-technical view on information systems and, at the same time, relates to ethical, social and political issues raised by information systems.

AIST provides a forum for academics and professionals to share the latest developments and advances in the knowledge and practice of these fields. It seeks new studies in many disciplines to foster a growing body of conceptual, theoretical, experimental, and applied research that could inform design, deployment and usage choices for information systems and technology within business and public organizations as well as households.

We call for papers covering a broad spectrum of topics which bring together sciences of information systems, information technologies, and social sciences, i.e., economics, management, business, finance, and education. The track bridges the diversity of approaches that contributors bring to the conference. The main topics covered are:

- Advances in information systems and technologies for business;
- Advances in information systems and technologies for governments;
- Advances in information systems and technologies for education;
- Advances in information systems and technologies for healthcare;
- Advances in information systems and technologies for smart cities; and
- Advances in information systems and technologies for sustainable development.

AIST invites papers covering the most recent innovations, current trends, professional experiences and new challenges in the several perspectives of information systems and technologies, i.e. design, implementation, stabilization, continuous improvement, and transformation. It seeks new works from researchers and practitioners in business intelligence, big data, data mining, machine learning, cloud computing, mobile applications, social networks, internet of thing, sustainable technologies and systems, blockchain, etc.

Extended versions of high-marked papers presented at technical sessions of AIST 2015-2021 have been published with Springer in volumes of Lecture Notes in Business Information Processing: LNBIP 243, LNBIP 277, LNBIP 311, LNBIP 346, LNBIP 380, LNBIP 413 and LNBIP 442.

Extended versions of selected papers presented during AIST 2022 will be published in Lecture Notes in Business Information Processing series(LNBIP, Springer).

- Data Science in Health, Ecology and Commerce (4th Workshop DSH'22)
- Information Systems Management (17th Conference ISM'22)
- Knowledge Acquisition and Management (28th Conference KAM'22)

TRACK CHAIRS

- **Ziemia, Ewa**, University of Economics in Katowice, Poland
- **Chmielarz, Witold**, University of Warsaw, Poland
- **Cano, Alberto**, Virginia Commonwealth University, Richmond, United States

PROGRAM CHAIRS

- **Chmielarz, Witold**, University of Warsaw, Poland
- **Miller, Gloria**, maxmetrics, Germany
- **Wątróbski, Jarosław**, University of Szczecin, Poland
- **Ziemia, Ewa**, University of Economics in Katowice, Poland

PROGRAM COMMITTEE

- **Ben-Assuli, Ofir**, Ono Academic College, Israel
- **Białas, Andrzej**, Instytut Technik Innowacyjnych EMAG, Poland
- **Byrski, Aleksander**, AGH University Science and Technology, Poland
- **Christozov, Dimitar**, American University in Bulgaria, Bulgaria
- **Dang, Tuan**, Posts and Telecommunications Institute of Technology, Vietnam
- **Dias, Gonçalo**, University of Aveiro, Portugal
- **Drezewski, Rafal**, AGH University of Science and Technology, Poland
- **Grabara, Dariusz**, University of Economics in Katowice, Poland
- **Halawi, Leila**, Embry-Riddle Aeronautical University, USA
- **Kania, Krzysztof**, University of Economics in Katowice, Poland
- **Kapczyński, Adrian**, Silesian University of Technology, Poland
- **Kluza, Krzysztof**, AGH University of Science and Technology, Poland
- **Kovatcheva, Eugenia**, University of Library Studies and Information Technologies, Bulgaria

- **Kozak, Jan**, University of Economics in Katowice, Poland
- **Ligeza, Antoni**, AGH University of Science and Technology, Poland
- **Ludwig, Andre**, Kühne Logistics University, Germany
- **Luna, Jose Maria**, University of Cordoba, Spain
- **Michalik, Krzysztof**, University of Economics in Katowice, Poland
- **Naldi, Maurizio**, LUMSA University, Italy
- **Nguyen, Thi Anh Thu**, The University of Danang, Vietnam
- **Pham, Van Tuan**, Danang University of Science and Technology, Vietnam
- **Rechavi, Amit**, Ruppin Academic Center, Israel
- **Rizun, Nina**, Gdansk University of Technology, Poland
- **Rollo, Federica**, University of Modena and Reggio Emilia, Italy
- **Rusho, Yonit**, Shenkar College of Engineering and Design, Israel
- **Saławun, Wojciech**, West Pomeranian University of Technology, Poland
- **Santiago, Joanna**, Universidade de Lisboa – ISEG, Portugal
- **Sikorski, Marcin**, Gdansk University of Technology, Poland
- **Solanki, Vijender Kumar**, CMR Institute of Technology(Autonomous), Hyderabad, TS, India
- **Taglino, Francesco**, IASI-CNR, Italy
- **Tomczyk, Łukasz**, Pedagogical University of Cracow, Poland
- **Webber, Julian**, Osaka University, Japan
- **Ziemba, Paweł**, University of Szczecin, Poland

A Blockchain-Based Self-Sovereign Identity Approach for Inter-Organizational Business Processes

Amal Abid, Saousssen Cheikhrouhou, Slim Kallel, and Mohamed Jmaiel

Abstract—Blockchain presents a promising and revolutionary technology for organizations’ collaboration, particularly for Inter-Organizational Business Processes (IOBP). It addresses the lack-of-trust problem thanks to its transparency and decentralized features. However, while the adoption of Blockchain technology can alleviate some of IOBP’s challenges, it does so at the expense of significant privacy issues. In fact, some process execution data, such as customers’ data or business secrets, cannot be shared across the collaborating organizations owing to regulatory restrictions such as the General Data Protection Regulation (GDPR). To address trust and privacy issues in IOBP, this paper presents a Blockchain-based Self-Sovereign Identity (SSI) approach. The SSI concept is combined with a registry proof smart contract to provide an efficient privacy-preserving solution. The proposed approach is applied to the pharmaceutical supply chain case study and implemented on the Ethereum Blockchain.

Index Terms—Blockchain, BPMN, IOBP, Self-Sovereign Identity

I. INTRODUCTION

BUSINESS processes have become the main factor of organizations to accomplish defined goals and to remain competitive in the dynamic marketplace. Collaboration between organizations, such as in supply chains, is considered essential in a business ecosystem in which organizations focus on their competitive strategy, perform only those operations for which they have expert skills, and enrich their services through partners and suppliers [1], [2].

In an Inter-Organizational Business Process (IOBP), independent organizations operate as collaborators and exchange messages to perform business transactions. This data exchange may be complicated, particularly when safety and confidentiality are intended to be first-class citizens. Collaborators expect to have access to complete process execution data and benefit from maintaining traceability. However, this is difficult to achieve in IOBPs since some process execution data such as customers’ data or business secrets cannot be shared across the collaborating parties owing to regulations and confidentiality restrictions (e.g. General Data Protection Regulation (GDPR) [3]). Furthermore, there is an inherently lack-of-trust problem as organizations are mutually untrusted and IOBP execution can be prone to disagreements on counterfeiting operations. Additionally, IOBP can hardly be established efficiently, since companies generally rely on settled

business processes and existing solutions. In fact, each party has managed information by building its own database. This has led to information silos, however, resulting in serious problems, particularly with respect to verification of data origins. Current systems use a centralized solution to organize their interoperability and cooperation. In this case, one of the actors will be the dominant partner in providing the solution and having access to the data. If instead, the parties choose a third-party solution, this would be costly and still prone to potentially exposing sensitive information.

Recently, Blockchain technology [4], [5] is proposed for IOBP execution to address the lack-of-trust problem, thanks to its nature as distributed, transparent and immutable ledger [6], [7], [8], [9]. While the adoption of Blockchain technology can alleviate some of these challenges, it does so at the expense of significant privacy issues. To overcome these problems, Blockchain can be leveraged in conjunction with Self-Sovereign Identity (SSI).

Self-Sovereign Identity (SSI) represents the recent evolution in identity management systems. In SSI systems, individuals have complete ownership and control over their data. These data that constitute an identity are known as Verifiable Credentials (VCs) and, unlike traditional systems, remain only with the individual. Verifiable credentials can also be owned by organizations as well as individuals. To protect privacy, SSI systems do not record transactions between interacting peers on ledger since they may include or reveal private information. Alternatively, the ledger is harnessed to verify claims using verifiable credentials.

In this line, some initial work is proposed for exploring the combination of Blockchain with Self-Sovereign Identity to address the issues highlighted above [10], [11], [12], [13], [14]. Unfortunately, these approaches focus particularly on the use of Blockchain-based SSI solutions for the customer side (business to customer (B2C)) and disregard their use between organizations (business to business (B2B)), and hence these approaches do not deal with IOBPs. Besides, these solutions do not address the lack of traceability concerns in SSI systems.

In this paper, we propose a Blockchain-based SSI solution for IOBP that ensures confidential inter-organization process execution while providing privacy-preserving traceability.

The main contributions of this paper can be summarized as follows :

- Propose an interoperable SSI interface between inter-organizational processes that exposes its functionality to the cooperative organizations through a common API.

A. Abid, S. Cheikhrouhou, S. Kallel, and M. Jmaiel are with ReDCAD, ENIS, University of Sfax, Sfax, Tunisia

The objective of the proposed SSI interface is to enable confidential collaboration between organizations by providing confidential end-to-end processes without exposing any sensitive data on-chain. It also supports the use of existing systems and databases while incorporating Blockchain technology as a reference for inter-organizational process interaction.

- Propose a Transaction (Tx) Registry Proof that maintains traceability in a private manner. In particular, this Registry Proof records the hash of transactions as well as Decentralized Identifiers (DIDs) of collaborating organizations in the Blockchain to ensure integrity. This can be used as a privacy-preserving trace to validate afterward that a collaborative task was performed in case of conflicts or audits.

To validate the feasibility of the proposed approach, we applied it to a pharmaceutical supply chain as a case study. In fact, the pharmaceutical supply chain is a vastly diversified and complex ecosystem, in which, the secure management of identity and private data is a typical concern. More Precisely, collaborating entities could identify their partners and interact with them using Verifiable Credentials for compliant transactions and information disclosures. This paper also provides an initial implementation and execution. An experimental evaluation shows that the implementation can achieve good results with low gas costs as well as low latency.

The remainder of this paper is organized as follows : Section II briefly introduces some concepts upon which our work is built. Section III explains the proposed approach. Section IV illustrates the pharmaceutical supply chain case study. Section V presents a proof-of-concept implementation. Section VI evaluates our approach. Section VII summarizes related work, and section VIII concludes and suggests future directions.

II. BACKGROUND

This section introduces the main concepts and definitions related to Self-Sovereign Identity (SSI).

Digital identities in today's world rely on the username-password combination method or the federated identity management method. Users are struggling to handle the growing number of passwords within the username-password combination method. Besides, they are vulnerable to password theft techniques including phishing, key-logging, viruses, and malware. They are also unable to efficiently transfer identity-related information from one account to another, and they must go through the hard registration procedure repeatedly, disclosing ID cards, driver's licenses, bank account details, and other personal information [15].

On the other hand, the federated identity management method attempts to alleviate some of these drawbacks with Single Sign-On platforms that transfer identity-related information between services that are linked to the platform. However, users are obliged to accept the terms and conditions because, otherwise, they will be unable to use the system. Additionally, during registration, users should disclose a significant amount of personal information, which causes

privacy issues. This personal information is not protected from unauthorized secondary use [13].

Self-sovereign Identity (SSI) represents the latest evolution and most current stage of digital identities, which is designed to address the issues of all previous stages. Thanks to SSI, users have full control over their data when using enterprises' systems. Besides, they can share only the required piece of information with their consent.

SSI's standards, architecture, and lifecycle are presented as follows.

A. SSI Standards

SSI is based on two standardized pillars. Decentralized Identifiers (DIDs) and their cryptographic counterparts, Verifiable Credentials (VCs), provide a decentralized and privacy-preserving form of digital identity.

- **Decentralized Identifier:** A decentralized identifier (DID) is an innovative type of globally unique identifier created by the World Wide Web Consortium (W3C) working group [16]. The DIDs approach has proven to be popular for associating a globally unique identifier to cryptographic keys and other interaction metadata necessary to prove control of the identifier.
- **Verifiable Credential:** A credential is a document that details the qualification, ability, or authority granted to an individual by a third party having the requisite authority or assumed ability. For example, a driver's license is used to prove that a person is capable of driving a vehicle, a university degree can be used to prove the education level of a person, and a government-issued passport permits people to travel between nations. These physical credentials may include information related to the identifier (e.g., identification number, photo), the issuing authority, particular attributes asserted by the issuing authority, and credential constraints. All the same information that a physical credential represents can be represented by a verifiable credential (VC), defined as a tamper evident credential that has authorship which can be cryptographically verified. The Verifiable Credential Data Model specification became a recommended standard by W3C in 2019 [16].

B. SSI Architecture

Figure 1 depicts the SSI architecture. In this architecture, there are three principal roles: issuer, verifier, and holder. They are briefly presented below:

- **Issuer:** An entity that creates claims within a VC about a subject. Such an entity can be organizations like governments, universities, but also private individuals or objects such as sensors. An issuer transfers VCs to holders.
- **Holder:** An entity that requests or receives VCs from issuers and maintains them in a credential repository/digital wallet. A holder may not always be the (credential) subject. For example, a parent (holder) holding VCs for its child (subject) or a friend (holder) obtaining a prescription at the pharmacy for its sick friend (subject).

Holders can also create Verifiable Presentations (VPs) from Verifiable Credentials and disclose them to a verifier.

- **Verifier:** An entity that intends to verify specific attributes or claims of a subject. It may receive these in the form of VP, which may include those claims from one or more VCs. However, holders have control at all times over which attributes are transferred to the verifier.

As a recent SSI development, DID Communication (DID-Comm) [17] presents an asynchronous encrypted communication protocol. It establishes a cryptographically secure channel for any two software agents (peers) to interact directly or via intermediary cloud agents. In DIDComm, peers who are parties to the connection are individually responsible for the generation of their DID, the key pairs in a DID document, and the subsequent key rotation or revocation of those keys. DIDComm uses information from the DID document, such as the public key and its associated endpoint, to exchange secure messages. It enables distinct entities to connect with each other in a peer-to-peer manner, eliminating the need for a middleman.

This credential exchange protocol supports zero-knowledge proof (ZKP) cryptography using the Camenisch-Lysyanskaya (CL) signature scheme, which enables credential holders to selectively reveal claims to verifiers without any linkage.

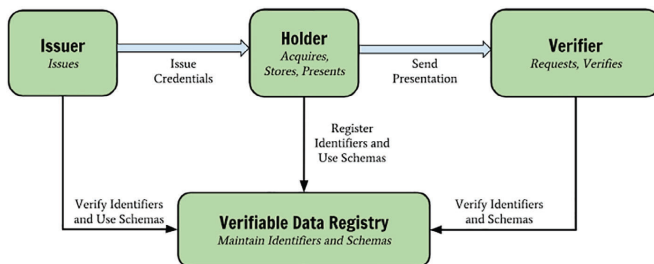


Fig. 1: SSI Architecture [16]

C. SSI Lifecycle Process

The lifecycle of a VC with Camenisch-Lysyanskaya (CL) signature which enables the zero-knowledge proof (ZKP) is detailed as follows:

- The issuer specifies the credential's schema, publishes the credential definition which indicates the intention to post a credential from schema X, signed using key Y, and with revocation strategy Z.
- The issuer generates a DID and correlates it with a public-private key pair. Afterwards, the issuer generates a DID document, signs it, and publishes it to the distributed ledger.
- The issuer collects all the information it intends to include in the credential, containing the information for each attribute specified in the schema defined previously. The issuer constructs the credential by creating a numeric representation of each field and then signs the numeric as well as the text formats of each of the claims using the CL Signature.

- The credential anchors a "link secret" that is known only to the holder (recorded in the holder's wallet), and when a credential is issued to the holder, it encapsulates a cryptographic promise to the "link secret" within another long number that the issuer serves as the credential ID. The "link secret" acts similarly to a watermark stamp. Therefore, the certificate's content is extremely difficult to falsify, proving that the holder owns the stamp and is able to create such a watermark.
- Once the holder owns the VC in his/her digital wallet, he/she can communicate with a verifier and may seek to prove a set of claims created by a specific issuer regarding a subject. The holder receives a request from the verifier for the type of credential it is looking for.
- The holder conducts certain calculations on the VC to prepare it for sharing in a proof presentation. The holder creates a new, never-before-seen credential wrapped inside a proof presentation. This later aggregates and discloses whatever attributes from issued credentials are requested, as well as any predicates, while hiding everything else. The 'proof' block of this new VC is a mathematical proof that the holder actually owns VCs signed by the appropriate issuer, containing the revealed attributes, and conforming to the specified schema.
- The proof also proves that the issuer has not revoked the credentials and that they are bound to the holder because the holder knows the "link secret" that was utilized at issuance. Afterwards, the verifier uses the information received from the holder in the form of a proof presentation to do certain calculations. It should cryptographically verify the validity of the proof. The verifier resolves the issuer's DID and identifies its public key. Then, using the issuer's public key, it validates the provided attributes. The presentation proof may comprise attributes from more than one credential. For each shared attribute, the verifier checks its corresponding credential schema, as well as the issuer's DID/DID document. It employs these two pieces of information to verify the presentation attribute. Each attribute statement in the proof presentation must follow this process. The verifier can be assured that all of the attributes are issued to the holder of the same "link secret".

III. BLOCKCHAIN-BASED SSI APPROACH FOR IOBP

This section provides a detailed description of the proposed approach. This description covers the proposed platform, the involved actors as well as the main process to ensure a secure IOBP communication (see Figure 2).

The main objective of the proposed Blockchain-based SSI solution is to ensure confidential inter-organization process execution while providing privacy-preserving traceability.

A. Platform components

The proposed platform includes three main components: an SSI interface, a private permissioned Blockchain and a Registry Proof smart contract which are described as follows.

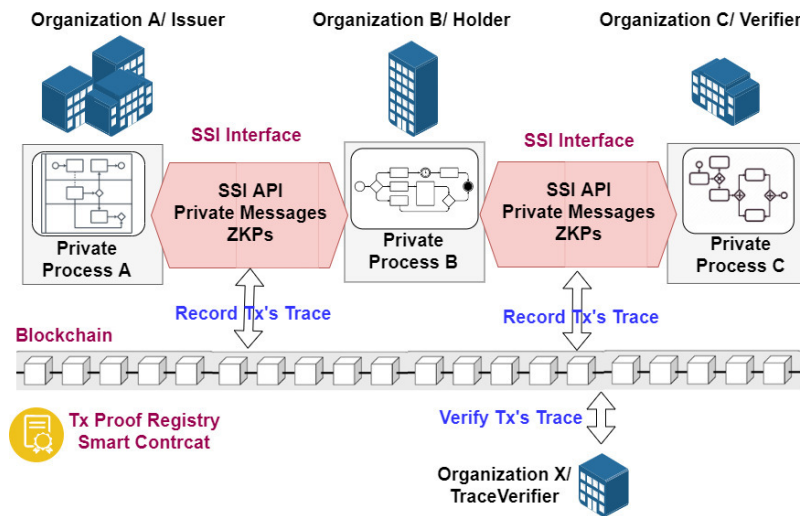


Fig. 2: Proposed approach Overview

- SSI Interface:** The SSI Interface between processes involved in the inter-organizational process presents modular libraries and APIs for Decentralized Identifiers (DID) and Verifiable Credentials (VC). It exposes its functionality to the cooperative organizations through a common interface API (i.e. RESTful API) to ensure interoperability. Therefore, collaborative organizations can keep using existing systems and databases while integrating this RESTful API as a common interface. The proposed SSI interface is aligned with the W3C specification and ensures JWT-based VC issuance, Selective Disclosure Request (SDR), and Verifiable Presentations (VP). Through the proposed SSI interface, each organization can present a DID document maintaining verification methods (i.e. public keys) and service endpoints (e.g., details of messaging service) that other organizations can use to establish interactions. Thus, before engaging in any formal activity in a relationship, two organizations should first mutually resolve each other's DID and acquire the interaction information preserved in the DID document.
- Private permissioned Blockchain:** The proposed platform relies on Ethereum [5] private permissioned Blockchain (i.e. Ethereum Private net) to allow only authorized users to access to the system by verifying their cryptographic keys. Furthermore, in the proposed platform, the Blockchain serves as a tamper-proof verifiable data registry as well as transactions' proof registry.
- Registry Proof Smart Contract:** The transactions' Proof Registry smart contract is proposed to record transactions' trace. Indeed, SSI systems do not offer a native way to record issuance and verification transactions on the Blockchain in order to ensure privacy and protect sensitive data. This can present a gap when audits are needed. Therefore, we propose to record only DIDs of interacting organizations as well as transaction hash to maintain a privacy-preserving trace. The data itself is stored locally off-chain for private process execution. While the proposed approach requires the use of a permis-

sioned Blockchain, a public Blockchain may also be used as a shared reference between collaborative organizations by recording the hash of each DID and the transaction hash.

B. Involved Actors

Many actors are involved in our proposed approach:

- Organizations:** In the proposed approach, each organization can play the role of issuer, holder or verifier when taking part in an inter-organizational process.
 - Issuer** (i.e., Organization A): When an organization issues a verifiable credential, its DID is associated with this credential for further verification. Here the issuer is trusted by different actors belonging to the permissioned Blockchain.
 - Holder** (i.e., Organization B): An organization has full control over its data including sensitive information and any request for accessing these data must necessarily require its confirmation. When performing a collaborative process, an organization can use the Selective Disclosure concept, during message exchange, to share selected pieces of information with interacting parties.
 - Verifier** (i.e., Organization C): An organization can verify the origin of exchanged data by checking digital signatures. Hence, it can confirm the validity and authenticity of shared verifiable credentials.
 - Trace Verifier:** A Trace Verifier is an authorized organization (i.e., Organization X) that has access to the trace provided by the Tx Registry Proof. It can check if an inter-organizational task/activity (i.e., exchange of message) was performed between collaborating parties. We note here that the TraceVerifier is distinct from the Verifier role belonging to the SSI concept.
- In the proposed solution architecture, each role may have multiple instances (i.e., multiple participating issuers, holders, verifiers, and trace verifiers).

C. Process flows

Figure 3 depicts the overview of the interaction between different actors involved in our proposed approach as a BPMN collaboration diagram.

The main interactions are as follows: An Organization A (i.e., Issuer) issues verifiable credentials to an Organization B (i.e., Holder) through message exchange in an inter-organizational collaboration. Afterwards, these verifiable credentials can be presented to an Organization C (i.e., Verifier) which confirms the validity and authenticity of data by verifying the signature of the issuer. The Tx Proof Registry records the transaction hash and the DIDs of interacting parties for both issuance and presentation activities. Consequently, an authorized Organization X (i.e., TraceVerifier) can check if the transaction between collaborating parties is performed.

Listing 1 illustrates the algorithm of the inter-organizational process of data exchange and storage.

This algorithm depicts a secure and private DIDComm between three cooperating organizations. An Organization A (i.e., Issuer) first encrypts and signs a message for Organization B (i.e., Holder). The signature and the cipher text are then sent through organization A's endpoint to organization B's endpoint. The authenticity of the message can be checked, by an Organization C (i.e., Verifier) before executing an IOBP, by resolving the DID and identifying whether it matches organization A's public key. All mentioned interactions are recorded in a privacy-preserving way on a Tx Proof Registry for further traceability by any authorized Organization X (i.e., Trace Verifier).

1. A process task exchanges DIDs with an Organization A (Org_A) (i.e. Issuer) to establish a DIDComm connection channel.
2. The Issuer employ the public key of the AES encryption scheme (pk_{aes}) to encrypt the data.
3. Data Issuer issues Verifiable Credential data (vc_{data}) to the process task with the (pk_{aes}) as an attribute of the credential.
4. Process task accepts and stores the Verifiable Credential in the Organization B (Org_B) (i.e. Holder) wallet.
5. The hash of the transaction ($trans_{proof}$) as well as Organization A (Org_A) and Organization B (Org_B) DIDs are stored on the Proof Registry.
6. Organization B (Org_B) exchange Verifiable Credential data with Organization C (Org_C) through a collaborative process within a DIDComm connection channel.
7. Organization C (Org_C) verifies the Proof Data by checking the signature and DID of the Issuer.
8. If proof data has been verified then
 - 9. Execute the current IOBP
 - 10. Store the hash of the transaction ($trans_{proof}$) as well as Organization B (Org_B) and Organization C (Org_C) on the Proof Registry.
11. An Organization X (Org_X) (i.e. TraceVerifier) can request transaction proof from the Proof Registry.
12. The TraceVerifier Evaluate the transaction proof and send the evaluation result for underlying organizations

Listing 1: Algorithm of SSI applied to IOBP

IV. CASE STUDY: PHARMACEUTICAL SUPPLY CHAIN

The pharmaceutical supply chain is a diversified and complex ecosystem, in which the secure management of identity and sensitive data is a typical issue.

The Drug Supply Chain Security Act (DSCSA) [18] enforces specific requirements on different types of stakeholders: manufacturers, repackagers, wholesale distributors, third-party logistics providers (3PLs), and pharmacies [19]. One such requirement is an extended Know Your Customer' regulation, which requires each entity to confirm that their partners are also authorized. In many situations, the regulation requires interactions between entities without any direct business relationship.

Therefore, to enable interoperability and trust, the pharmaceutical supply chain community has harnessed the power of Blockchain and Decentralized Identifiers (DIDs). Together, they provide all parties with a 'single source of truth' to address challenges, such as master data management and counterfeit detection. At a more essential level, different entities must be able to identify their partners and interact with them using Verifiable Credentials for compliant transactions and information disclosures.

The specific goals of this case study are: (i) authentication of a verification request with a verifiable credential, and (ii) enhanced verification between pharmacies and manufacturers.

Figure 4 shows a simplified model of the pharmaceutical supply chain as a BPMN collaboration diagram. The diagram contains seven pools, one for each involved parties: Raw Material Manufacturer, Pharmaceutical Manufacturer, Hospital Pharmacy, Hospital Healthcare Professional, Patient, Tx Proof Registry, Higher Authority of Drugs, and Pharmaceutical Industry (HADPI).

This diagram depicts many examples of data exchange between cooperative organizations, in which we use verifiable credentials' issuance and verification, such as *Raw Material Credentials*, *Pharmaceutical Credentials*, and *Product Credentials*.

We use colors to explain different roles and interactions. For example, the red color designates the issuance of *Raw Material Credentials* by the *Raw Material Manufacturer* which acts as an Issuer. These credentials are issued to the *Pharmaceutical Manufacturer* which plays the role of Holder. This issuance transaction is recorded in the Tx Proof Registry after its completion. The yellow color highlights how the *Pharmaceutical Manufacturer* can become an Issuer of the *Pharmaceutical Credentials*, while the *Hospital Pharmacy* acts as a Holder. The same interaction is performed between the *Hospital Pharmacy* and the *Hospital Healthcare Provider* (green color). Finally, the *Hospital Healthcare Provider* prepares a Verifiable Presentation to the *Patient* that contains *Pharmaceutical Product Claims* (blue color). Here the *Hospital Healthcare Provider* plays the role of a Holder while the *Patient* acts as a Verifier.

All issuance and verification transactions are recorded on the Tx Proof registry. Consequently, the HADPI can play the role of Trace Verifier, and thus check if transactions are performed between interacting parties.

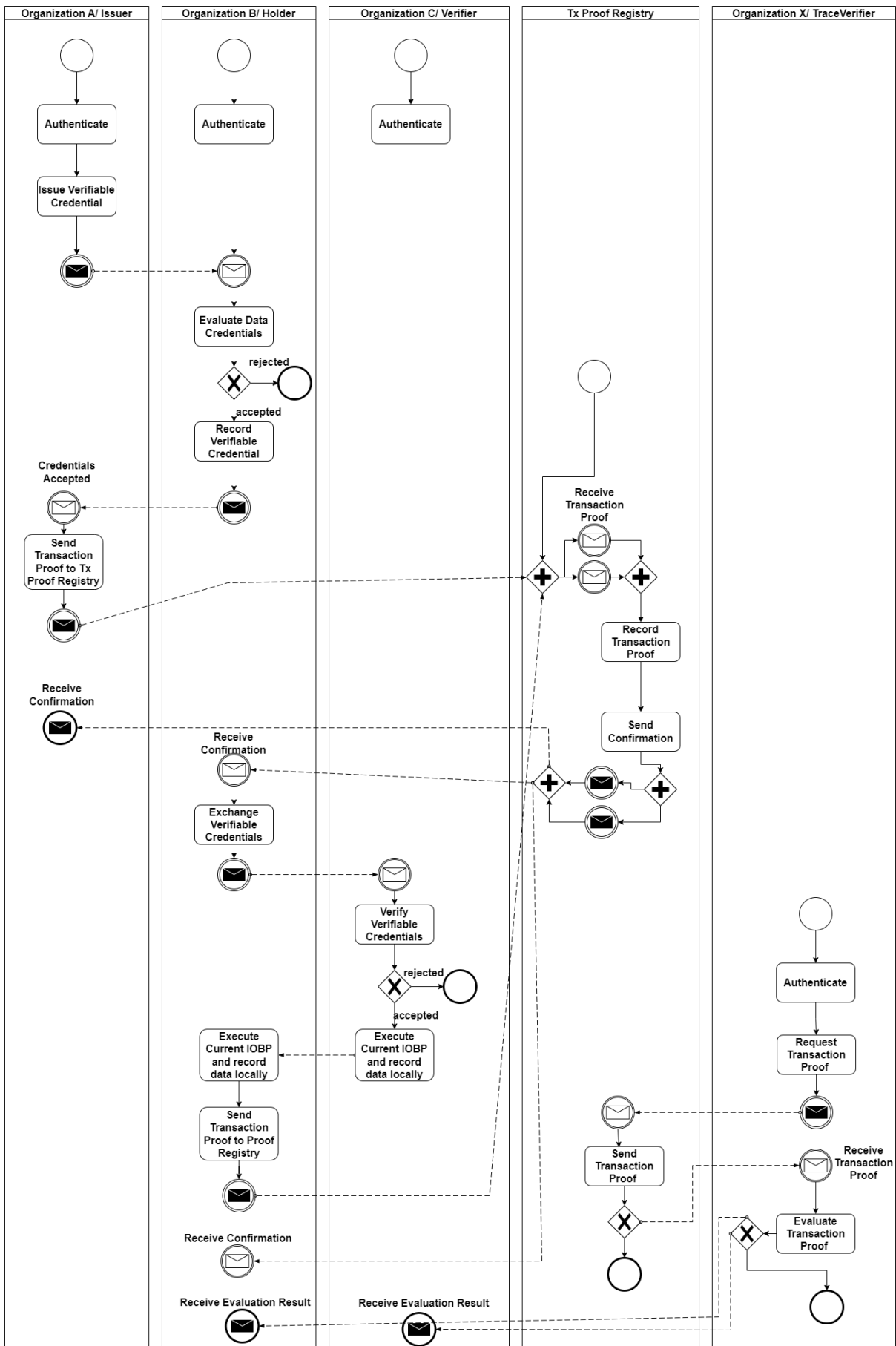


Fig. 3: Generalized BPMN diagram

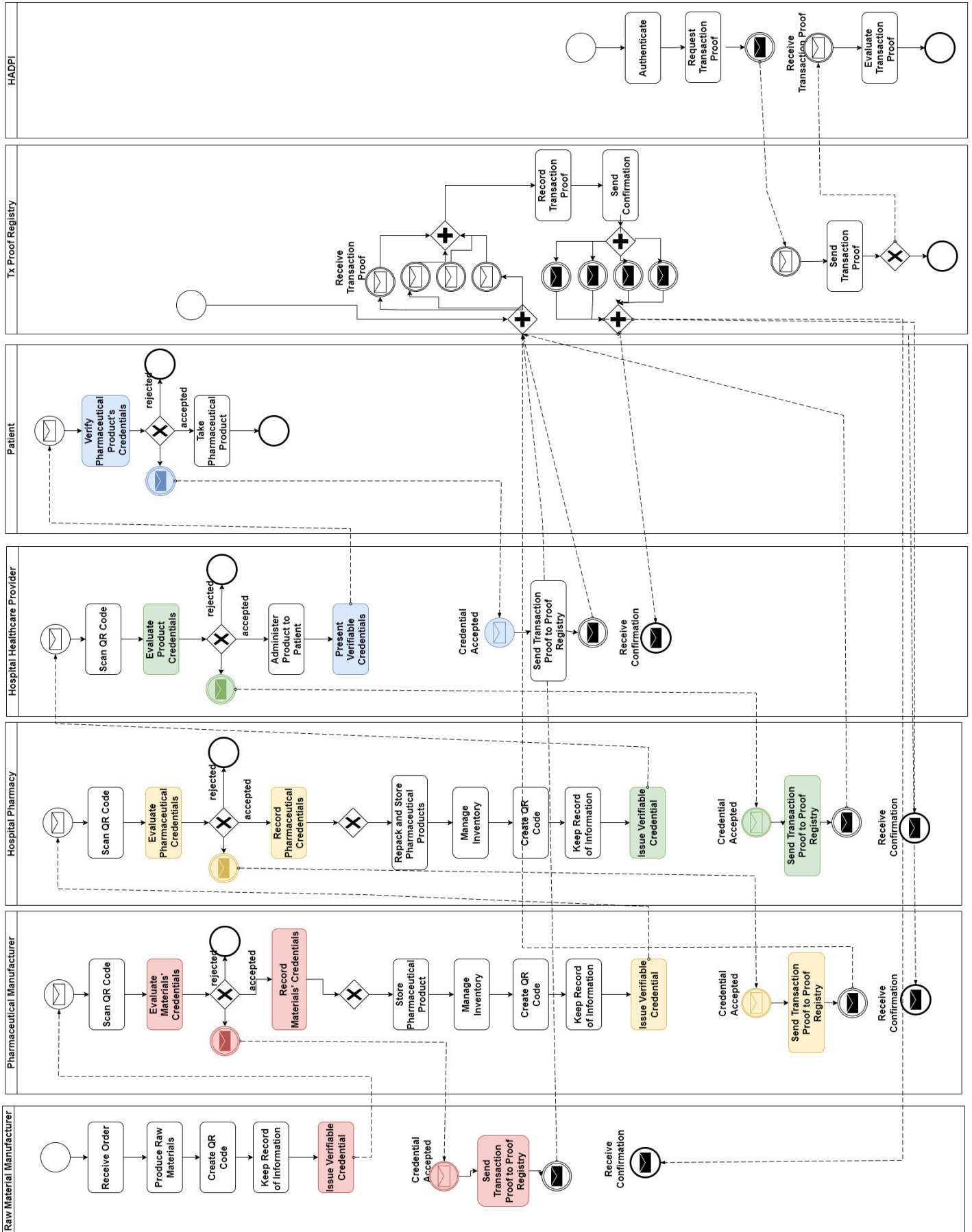


Fig. 4: Pharmaceutical Supply Chain's BPMN diagram

V. IMPLEMENTATION

This section presents the implementation of the proposed Blockchain-based SSI approach for IOBP. In particular, we detail the implementation of both the SSI interface and the Tx Proof Registry smart contract.

A. SSI interface

In order to implement the SSI interface, we use Veramo [20], an open-source set of modular libraries and APIs for SSI and verifiable credentials. Veramo exposes its functionalities to the cooperative organizations through a common RESTful API. Therefore, collaborative organizations can keep using existing systems and databases, while integrating Veramo RESTful API as a common interface. Additionally Veramo's API is aligned with the W3C specification, and it supports the creation of Ethereum-based and web-based DIDs as well. Besides, the SSI interface ensures JWT-based VC issuance, Selective Disclosure Request (SDR) and Verifiable Presentations (VP).

The first step towards using and accessing the methods of Veramo API is to create a Veramo Agent and export it in the 'VeramoSetup.ts'. As a result, this Agent can be imported and used in the proposed SSI interface. Listing 2 shows an excerpt of the implementation of the Veramo Agent.

```

1 // Core interfaces
2 import { createAgent, IDIDManager } from "@veramo/core";
3 // Core identity manager plugin
4 import { DIDManager } from "@veramo/did-manager";
5 // Credential Issuer
6 import { CredentialIssuer, ICredentialIssuer } from "
7   @veramo/credential-w3c";
8 export const veramoAgent = createAgent<IDIDManager &
9   IKeyManager & IDataStore & IResolver & ... >({ plugins:
10   [ new KeyManager({
11     store: new KeyStore(dbConnection, new SecretBox(secretKey)
12     ),
13     kms: {local: new KeyManagementSystem(), },
14   }]),
15   new DIDManager({ store: new DIDStore(dbConnection),
16     defaultProvider: "did:key",
17     providers: { "did:key": new KeyDIDProvider({ defaultKms:
18     "local", })})
19   }]),
20   new DIDResolverPlugin({ resolver: new Resolver({ key:
21     getDidKeyResolver().key, getUniversalResolverFor(["io",
22     "elem", "sov"], )}),
23   }]),
24   new CredentialIssuer(),
25   new MessageHandler({ ... })),
26   });

```

Listing 2: Excerpt of the Creation of Veramo Agent

To construct the agent, all required plugins must be imported as libraries (Listing 2 lines 1-6) and taken into consideration when the object is initialized (Listing 2 lines 7-18).

After its creation, the Veramo Agent provides basic methods for creating Verifiable Credentials, Verifiable Presentations, verifying messages like JWT credentials and sending presentation requests as Selective Disclosure Requests.

Listing 3 shows an excerpt of the creation of a VC using the Veramo Agent.

```

1 async issueVerifiableCredential(body:
2   IssueCredentialRequest, toWallet: boolean): Promise<
3   IssueCredentialResponse> {
4   try {

```

```

3   body.credential.issuer = {id: body.credential.issuer.
4     toString()};
5   const save: boolean = body.options.save?body.options.save:
6     false;
7   const credential: W3CCredential = body.credential;
8   const verifiableCredential: W3CCredential = await
9     veramoAgent.createVerifiableCredential({ save: false,
10     credential, proofFormat: "jwt", });
11 // Prepare response
12 const result: IssueCredentialResponse = { credential:
13   verifiableCredential, };
14 if (toWallet) {
15   try { // Send VC to another Veramo agent
16     const msg = await veramoAgent.sendMessageDIDCommAlpha1({
17       save: true,
18       data: { from: verifiableCredential.issuer.id, to:
19         verifiableCredential.credentialSubject.id, type: "jwt",
20         body: verifiableCredential.proof.jwt, }, });
21     result.sent = true;
22     return result;
23   } catch (error) {
24     return error; }
25 }
26 return result;
27 } catch (error) {
28   return error; }
29 };

```

Listing 3: Excerpt of the Creation of a VC using the Veramo Agent

The credential object is prepared first (Listing 3 Lines 3-5), and then transformed to a VC using Veramo Agent's methods (Listing 3 line 6). Afterwards, the API response is prepared with the newly created VC (Listing 3 lines 8-10). If an error occurs during issuance, the error is returned as a response to the requester. Alternatively, if the API request specify that the VC should be sent directly to the DID's agent through the messaging endpoint (Listing 3 lines 11-16), it would be handled using the "sendMessageDIDCommAlpha1()" method.

Note that the source code of the proposed SSI interface is available on Github in [21].

B. Tx Proof Registry Smart Contract

The proposed Tx Proof Registry Smart Contract provides a privacy-preserving trace of SSI transactions. Indeed, SSI issuance and verification transactions need to be persistently recorded in a private manner for further verification (i.e. check if these transactions are actually performed). An excerpt of the proposed smart contract is presented in listing 4. It records both DIDs of interacting organizations as well as transaction hash.

```

1 event recordTrace ( address indexed sender , address
2   indexed receiver ,
3   bytes32 txHash )

```

Listing 4: Excerpt of the Proposed Tx Proof Registry Smart Contract

While the proposed approach requires the use of a permissioned Blockchain, a public Blockchain may also be used as a shared reference between collaborative organizations. In this case, we recommend recording the hash of each DID instead of directly recording the DIDs on-chain (see listing 5).

```

1 event recordTrace ( bytes32 indexed sender , bytes32
2   indexed receiver ,
3   bytes32 txHash )

```

Listing 5: Excerpt of Tx Proof Registry Smart Contract for public Blockchain

VI. EVALUATION

This section evaluates the proposed approach and shows its feasibility and efficiency for adoption within a real-world environment including both financial cost and response time.

A. Financial Cost

Transactions on Ethereum Blockchain are subject to a certain fee. Ethereum employs a unit known as gas to calculate the amount of operations required to complete a task such as deploying a smart contract or executing an ABI function. It is always necessary to estimate gas consumption when implementing a smart contract in order to avoid unexpected costs. Therefore, storing data directly on-chain suffers not only from privacy issues but also from being costly.

In the proposed Tx Proof Registry smart contract, only the DIDs of interacting organizations and the hash of the transaction are recorded on-chain, the data itself is stored locally off-chain for private process execution. Table I shows the transaction cost for the execution of 'recordTrace' function as well as the deployment of the Tx Proof Registry smart contract.

TABLE I: Operations' Gas Cost

Operation	Gas
Tx Proof Registry Smart Contract Deployment	362,525
Record Trace ABI call	64,384

B. Response time

The proposed approach is dependent on Ethereum Blockchain's latency. In fact, despite the reduction of data-size recorded on-chain, storing DIDs and transactions' hashes on-chain leads to some overheads. These overheads can be tested by sending simultaneous requests to the Tx Proof Registry Smart Contract. Figure 5 depicts the completion time in seconds (Y axis) of the "TraceRecord" operation where we send between 1 and 800 simultaneous requests (X axis).



Fig. 5: Response Time Evaluation

To sum up, the experimental evaluation shows that the implementation can achieve good results with low gas costs as well as low latency.

VII. RELATED WORK

This section provides an overview of the existing solutions for secure inter-organizational collaborations.

Authors in [10] proposed a distributed Private Data System (PDS) to achieve self-sovereign storage and sharing of private data between multiple organizations through executable choreographies. The users have complete control over their private data and are allowed to share and revoke access to organizations at any time. However, PDS does not leverage the power of Blockchain technology to address consensus problems in a distributed environment. Instead, the PDS's system is composed of nodes spread across the entire Internet managing local key-value databases. This could present a complex infrastructure and require some effort, thus may be inefficient for individuals.

Another interesting work that exploited the strength of the Blockchain technology to ensure privacy-preserving inter-organizational collaborations is proposed in [11]. Authors presented, ID-Service, a platform for designing, implementing, and executing Cross-Organization Workflows' services. It adheres to the concept of security by design in terms of trust, accountability, non-repudiation, and the system's capacity to offer forensic proof of workflow traces, critical actions, and actors' responsibilities and to maintain these features during execution. However, ID-Service does not implement the Self-Sovereign Identity (SSI) model. Consequently, it does not afford any flexibility for identity's self-possession and control.

The SSI approach has mostly been explored in the field of security, privacy, and distributed systems, with little attention paid to information systems research and process perspective. At present, there are only a few academic papers on the application of SSI in business scenarios. They include the application of SSI in know-your-customer processes in banking [22], remote management of industrial equipment [23], payback programs in retail [12], student exchange [24], e-petitions [25], access to public health services [26], assigning medical information to persons without regular identity, e.g. to combat COVID-19 [27]. The majority of these studies represent typical business processes that consider in particular the Consumer-to-Business relationship and omit dealing with inter-organizational collaborations (i.e. Business-to-Business (B2B)). For instance, authors in [12] introduced an SSI-based system for incentivized and self-determined customer-to-business data sharing in a local economy context. Here consumers are not only the owners of their data, but also they can choose what to share and in which granularity to trade their data for financial rewards. In the same direction, authors in [13] presented SSI as a solution to deal with privacy-preserving licensing of individual-controlled data to avoid unauthorized secondary customers' data usage.

The majority of the aforementioned approaches use the SSI concept to ensure data privacy and omit to propose a privacy-preserving solution to trace SSI transactions for further audit requirements. This is presented in a very broad sense in [14]. Authors proposed the concept of a proof registry through a set of technical components, data structures, and process flows, that ensures that proof data can be validated in

case of disputes or audits. However, authors did not provide any implementation nor an example of a smart contract to illustrate/show how the traceability is performed. Besides, authors do not provide a solution for inter-organizational process execution.

Unlike all cited previous work, the proposed approach provides a Blockchain-based SSI solution for inter-organizational processes. Particularly, it proposes an interoperable SSI interface between collaborating organizations as well as a privacy-preserving proof registry for further audits and verification.

VIII. CONCLUSION

In this paper, we proposed a Blockchain-based SSI approach for IOBP. It ensures confidential inter-organization process execution, particularly inter-organization message exchange without exposing any sensitive information on-chain. The SSI concept is combined with a proof registry smart contract to provide a privacy-preserving trace for further audit verification.

In future work, we aim to enhance the proposed proof registry to enable enriched analysis on private data for authorized organizations.

REFERENCES

- [1] R. Wehlitz, F. Jauer, I. Rößner, and B. Franczyk, "Increasing the reusability of iot-aware business processes." in *Proceedings of the Conference on Computer Science and Information Systems (FedCSIS)*, 2020, pp. 17–22.
- [2] M. Nizioł, P. Wisniewski, K. Kluza, and A. Ligeza, "Characteristic and comparison of uml, bpmn and epc based on process models of a training company," in *Proceedings of the Conference on Computer Science and Information Systems (FedCSIS)*, vol. 26, 2021, pp. 193–200.
- [3] P. Voigt and A. Von dem Bussche, "The eu general data protection regulation (gdpr)," *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, vol. 10, no. 3152676, pp. 10–5555, 2017.
- [4] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," *Cryptography Mailing list*, 2008.
- [5] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger," *Ethereum project*, vol. 151, pp. 1–32, 2014.
- [6] O. López-Pintado, L. García-Bañuelos, M. Dumas, I. Weber, and A. Ponomarev, "Caterpillar: A business process execution engine on the ethereum blockchain," *Software: Practice and Experience*, vol. 49, no. 7, pp. 1162–1193, 2019.
- [7] A. B. Tran, Q. Lu, and I. Weber, "Lorikeet: A model-driven engineering tool for blockchain-based business process execution and asset management." in *Proceedings of the BPM Demo Track and BPM Dissertation Award co-located with the International Conference on Business Process Modeling (BPM)*, 2018, pp. 56–60.
- [8] O. López-Pintado, M. Dumas, L. García-Bañuelos, and I. Weber, "Controlled flexibility in blockchain-based collaborative business processes," *Information Systems*, p. 101622, 2020.
- [9] A. Abid, S. Cheikhrouhou, and M. Jmaiel, "Modelling and executing time-aware processes in trustless blockchain environment," in *Proceedings of the International Conference on Risks and Security of Internet and Systems*, 2019, pp. 325–341.
- [10] S. Alboae and D. Cosovan, "Private data system enabling self-sovereign storage managed by executable choreographies," in *Proceedings of the International Conference on Distributed Applications and Interoperable Systems (IFIP)*. Springer, 2017, pp. 83–98.
- [11] L. Argento, F. Buccafurri, A. Furfaro, S. Graziano, A. Guzzo, G. Lax, F. Pasqua, and D. Saccà, "Id-service: A blockchain-based platform to support digital-identity-aware service accountability," *Applied Sciences*, vol. 11, no. 1, p. 165, 2020.
- [12] K. Wittek, L. Lazzati, D. Bothe, A.-J. Sinnaeve, and N. Pohlmann, "An ssi based system for incentivized and selfdetermined customer-to-business data sharing in a local economy context," in *Proceedings of the IEEE European Technology and Engineering Management Summit (E-TEMS)*. IEEE, 2020, pp. 1–5.
- [13] M. Kang and V. Lemieux, "A decentralized identity-based blockchain solution for privacy-preserving licensing of individual-controlled data to prevent unauthorized secondary data usage," *Ledger*, vol. 6, 2021.
- [14] V. Lemieux, A. Voskobojnikov, and M. Kang, "Addressing audit and accountability issues in self-sovereign identity blockchain systems using archival science principles," in *Proceedings of the IEEE Computers, Software, and Applications Conference (COMPSAC)*. IEEE, 2021, pp. 1210–1216.
- [15] J. Sedlmeir, R. Smethurst, A. Rieger, and G. Fridgen, "Digital identities and verifiable credentials," *Business & Information Systems Engineering*, vol. 63, no. 5, pp. 603–613, 2021.
- [16] V. C. W. Group, "Decentralized identifiers (dids) v1.0. world wide web consortium (w3c) (2020) [online]. available: <https://www.w3.org/tr/vc-imp-guide/>."
- [17] DID, "Didcomm messaging [online]. available: <https://github.com/decentralized-identity/didcomm-messaging/>."
- [18] DSCSA, "Drug supply chain security act (dcsa) [online]. available: <https://www.fda.gov/drugs/drug-supply-chain-integrity/drug-supply-chain-security-act-dcsa/>."
- [19] V. Dods and B. Taylor, "A proposal for decentralized, global, verifiable health care credential standards grounded in pharmaceutical authorized trading partners," *Blockchain in Healthcare Today*, 2021.
- [20] Veramo, "Performant and modular apis for verifiable data and ssi [online]. available: <https://veramo.io/>."
- [21] A. Abid, "Ssi4iobp [online]. available: <https://github.com/amal-abid05/ssi4iobp/>."
- [22] R. Soltani, U. T. Nguyen, and A. An, "A new approach to client onboarding using self-sovereign identity and distributed ledger," in *Proceedings of the IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*. IEEE, 2018, pp. 1129–1136.
- [23] P. C. Bartolomeu, E. Vieira, S. M. Hosseini, and J. Ferreira, "Self-sovereign identity: Use-cases, technologies, and challenges for industrial iot," in *Proceedings of the IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, 2019, pp. 1173–1180.
- [24] P. Kavassalis, "Designing an academic electronic identity management system for student mobility using eidas eid and self-sovereign identity technologies," 2020.
- [25] R. Karatas and I. Sertkaya, "Self sovereign identity based e-petition scheme," *International Journal of Information Security Science*, vol. 9, no. 4, pp. 213–229, 2020.
- [26] D. W. Chadwick, R. Laborde, A. Oglaza, R. Venant, S. Wazan, and M. Nijjar, "Improved identity management with verifiable credentials and fido," *IEEE Communications Standards Magazine*, vol. 3, no. 4, pp. 14–20, 2019.
- [27] A. Abid, S. Cheikhrouhou, S. Kallel, and M. Jmaiel, "Novidchain: Blockchain-based privacy-preserving platform for covid-19 test/vaccine certificates," *Software: Practice and Experience*, vol. 52, no. 4, pp. 841–867, 2022.

Students' Online Behaviour in the Time of the COVID-19 Pandemic: Insights from Poland and Ukraine

Dariusz Dymek, Mariusz Grabowski,
Grażyna Paliwoda-Pękosz
Cracow University of Economics
Rakowicka St 27,
31-510 Kraków, Poland

{dymekd, grabowsm, paliwodg}@uek.krakow.pl

Svitlana Didkivska
Cracow University of Economics
Rakowicka St 27,
31-510 Kraków, Poland
d2023@student.uek.krakow.pl

Tetiana Anatoliivna Vakaliuk
Zhytomyr Polytechnic State University
Chudnivska St, 103,
10005, Ukraine
tetianavakaliuk@gmail.com

Abstract—The COVID-19 pandemic forced universities to rapidly switch to distance learning, while simultaneously accelerating changes that might facilitate a more inclusive model of education. The goal of the research was to investigate the multi-dimensional aspects of learning online, including students' gender, age, and culture. The research results, based on 1562 survey responses from Polish and Ukrainian students, suggest that there are still differences between men and women as far as digital competences are concerned that might have their background in the traditional perception of gender roles. Besides, the pandemic has accelerated the processes of hardware and software enhancement in order to facilitate online learning, especially in Ukraine, which is mitigating technological exclusion. The possibility of accessing learning resources online, might allow people with lower economic status or women to continue education. The research results might be helpful in the development of future educational policies at Polish and Ukrainian universities.

I. INTRODUCTION

The COVID-19 pandemic has triggered research concerning distance education and the future of education. The main investigated paths include: organisation of online learning [1], [2], students' and teachers' attitude towards distance learning [3], and students' wellbeing [4]. Distance learning has the potential to deepen the equality, fairness, and inclusion among the members of the society. It should be noted that distance learning also gives the opportunity for wider access to education for unprivileged people, i.e. inhabitants of rural areas, poor members of some ethnic groups and/or people with disabilities [5]. However, to the best of our knowledge the topic of opportunities that have arisen during the COVID-19 pandemic for shaping the more inclusive future education has not yet been thoroughly investigated. In recent years, the number of Ukrainian students in Poland has been growing constantly. From the perspective of teaching organisations, it is of great importance to investigate whether there are any differences between Polish and Ukrainian students and between students of different gender or age in case of their online behavioural pattern, as it might influence their attitude to learning in general, and to distance learning in particular. This was the main motivation for our research. More

specifically, we would like to answer the research question: Are there any relationships between gender, age, country and online behaviour before and during the pandemic? In particular we are interested whether there are any differences in: (RQ1) perceived student wellbeing? (RQ2) hours spent online? (RQ3) social behaviour patterns (social cycle)? (RQ4) attitudes towards online communication? (RQ5) attitudes towards online education?

In order to answer the research questions, a joint project was undertaken by the faculty of the Cracow University of Economics (CUE), Poland and the Zhytomyr Polytechnic State University (ZPSU), Ukraine.

II. RESEARCH BACKGROUND

The pandemic forced universities to switch to distance learning and thus provided a unique opportunity to experience this type of learning even by people who previously avoided it. Belan [6] has analysed the attitudes of Polish and Ukrainian students towards remote learning and teaching technologies in training teachers of vocational education. He points out that Polish experience may be used as a benchmark for Ukraine. Basing on that premise the author proposes the modernization of the Ukrainian educational system. Klapkiv and Dluhopolska [7] discuss the challenges and opportunities during the quarantine caused by COVID-19 in Poland and Ukraine. It was indicated that the Ukrainian education system was largely unprepared to tackle these challenges due to the bureaucracy, the process rather than result orientation, educational conformism, and the lack of motivation of the main stakeholders.

The influence of distance education on the mental health of students has been already discussed in research works. For example, the relationship between the neuroticism and the pandemic of Polish and Ukrainian students was examined by Długosz and Kryvachuk [8]. The research indicates that high levels of neuroticism were observed among 61% of respondents from Poland and 47% from Ukraine. The research has also indicated that Ukrainian students better cope with

TABLE I
RESPONDENTS' STRUCTURE

		Poland		Ukraine	
		No.	%	No.	%
Gender	Female	643	64%	234	42%
	Male	357	36%	305	55%
	Not specified	5	0%	18	3%
Age	Young: < 20	143	14%	329	59%
	Medium: 20-24	727	72%	213	38%
	Older: 24+	135	13%	15	3%

quarantine and have better mental health. The problem of mental health deterioration during the pandemic has also been investigated in a broader perspective by Ochnik et al. [9]. The research outcomes indicate that the state of students' mental health is alarming and higher educational institutions (HEIs) should provide psychological support for them. There are also research works that explore a broader perspective of distance education, e.g. the problem of the limitations concerning the continuation of university education [4].

III. RESEARCH METHODS

The research is part of a project undertaken by the faculty of CUE and ZPSU. The goal was to investigate multi-dimensionally the students' perspective of distance learning in the time of the pandemic. Preliminary results of the research that looked at only one country's perspective were published in [10] and [11], with Polish and Ukrainian data respectively. This paper extends these studies by comparing data and providing a multi perspective analysis of respondents' viewpoints, including gender, age, and country.

We gathered the data in May and June 2021 using a questionnaire (see Appendix). In Poland, the survey was sent to students of CUE, mainly representing business studies. 1005 questionnaires were received giving a response rate of 8% (Cochran Formula: the margin of error is equal to 3% with a confidence level of 95%). In Ukraine, the questionnaire was distributed among students of seven universities: Zhytomyr Polytechnic State University, National University of Life and Environmental Sciences, Uman State Pavel Tychyna Pedagogical University, Melitopol State Pedagogical University, Drohobych State Pedagogical University, National Pedagogical University, and Kryvyi Rih State Pedagogical University. In total, we received 557 responses from Ukrainian students. We analysed the data using descriptive statistics.

The respondents' structure presents Table I. As far as respondents' digital competences are concerned, about 40% of respondents from Poland and Ukraine assessed their skills as average, however only 2.4% of Poles assess their skills as below average, while 17% of Ukrainians assess their skills as such. There were visible differences between age groups. It is especially apparent in the Young group, where the results are 0.9% and 13.5%, respectively. Hence, it seems that the Ukrainian student population is much more diversified as far as digital competences are concerned. In both countries men

TABLE II
AVERAGE SELF-PERCEPTION OF WELLBEING

		Poland	Ukraine
Total		2.75	2.78
Gender	Female	2.72	2.74
	Male	2.79	2.82
Age	Young	2.73	2.77
	Medium	2.66	2.78
	Older	3.23	3.00

1-severe deterioration, 2-deterioration, 3-neutral, 4-improvement, 5-considerable improvement

assessed their digital competences higher than women. About 9% of Ukrainians reported constant or frequent technical problems, whereas in Poland only 3%. This indicates the differences between technical environments in Poland and Ukraine. In both countries, women assess their technical conditions slightly better, similarly to the Older group of respondents.

In order to capture the possible changes in students' online behaviour, we assessed the number of "friends", both with respect to online and face-to-face contacts before and during the pandemic. We defined "friends" as people with whom respondents actively maintain contact in the private sphere.

IV. RESULTS

RQ1: Students' Wellbeing. The deterioration in students' wellbeing is visible among both Polish and Ukrainian students at a similar level (Table II). However, the Older Ukrainian group did not notice changes in wellbeing (mean: 3.0), while the Older Polish group reported a slight increase in well-being (mean: 3.23).

RQ2: Hours Spent Online. There was a difference in the time spent online between Polish and Ukrainian students before the pandemic: Ukrainian students on average spent six hours more online per week than Polish students. These differences were visible in all categories. It should be noted that during the pandemic the numbers of hours spent online were almost equal in Poland and Ukraine, respectively 36 and 35 hours per week. However, in both respondent populations, women spent less hours online than men, both before and during the pandemic. Similarly, the number of hours spent online in relation to studies before the pandemic was higher in Ukraine than in Poland, the difference visible in all categories.

Table III shows that online studies contributed mostly to the increase in the number of hours online. In percentage change, 100% means that all of the increase is related to online learning. The values for Poland are close to 100%, hence it seems that the entire change was related to online learning. In the case of Ukraine, except for the Young group, the values are quite distant from 100%, for the Medium group it is 129%, which can be interpreted that online learning forced the resignation from other online activities, while in the case of the Older group, the increase in online activity was related to activities other than learning.

TABLE III
PROPORTION OF THE TOTAL ONLINE ACTIVITY INCREASE ATTRIBUTED TO ONLINE STUDIES AND THE CHANGE OF THE PERCENTAGE OF ONLINE LEARNING IN TOTAL ONLINE ACTIVITIES

		Increase		Change	
		Poland	Ukraine	Poland	Ukraine
Total		97%	110%	17%	8%
Gender	Female	98%	97%	16%	7%
	Male	97%	123%	15%	9%
Age	Young	104%	101%	20%	6%
	Medium	96%	129%	16%	12%
	Older	103%	37%	15%	-8%

In the time spent online related to studies, it is worth noting that in Poland there was an increase in each group (although the increase gets smaller with age), which means that online learning was a significant factor in the increase in time spent online. A similar situation was observed in the Ukrainian Young and Medium groups, but it is interesting that in the Older group we have a decrease, which means that the increase in time spent online was mostly not due to online learning, but other online activities. This can be related to the types of contacts – in the Older Ukrainian group the largest decrease in face-to-face contacts can be observed, with the largest increase in online contacts, which directly translated into increased online activity related not to studies, but to maintaining social contacts.

The share of online learning in the total time spent online increased both in Poland and Ukraine, but it seems that Ukrainians spent proportionally more time on learning online before the pandemic (Table III).

RQ3: Social Pattern Behaviour. There are visible differences in the average number of friends online between Polish and Ukrainian respondents: the average number of contacts online decreased in Poland from 13 to 12 and increased in Ukraine from 11 to 14. On average, the number of face-to-face friends decreased in both populations, however this change was more drastic in the case of Polish respondents (decrease from 16 to 8; among Ukrainian respondents the decrease from 10 to 8) (Table IV shows the changes). Polish respondents keep in touch with online contacts much more intensively than their Ukrainian counterparts. In both respondent groups the increase in constant online communication is visible during the pandemic but more considerable in the case of Polish respondents (16% increase).

RQ4: Attitudes towards Online Communication. The pandemic contributed to a more favourable perception of online communication among Ukrainian respondents (Fig. 1). Although it is difficult to notice any global differences between Poland and Ukraine in terms of age, it is interesting that in terms of communication “constantly” the increase in Ukraine is slight, while in Poland it is significant. There is an increase in the number of online contacts in Ukraine. The number of contacts is growing but their intensity is smaller, unlike in Poland, where we observed a decrease in the number of

TABLE IV
CHANGE IN A NUMBER OF ONLINE AND FACE-TO-FACE FRIENDS

		Online friends		Face-to-face friends	
		Poland	Ukraine	Poland	Ukraine
Total		-1	3	-8	-2
Gender	Female	-1	3	-9	-2
	Male	0	2	-7	-1
Age	Young	-2	3	-7	-2
	Medium	-1	2	-9	-1
	Older	2	4	-7	-5

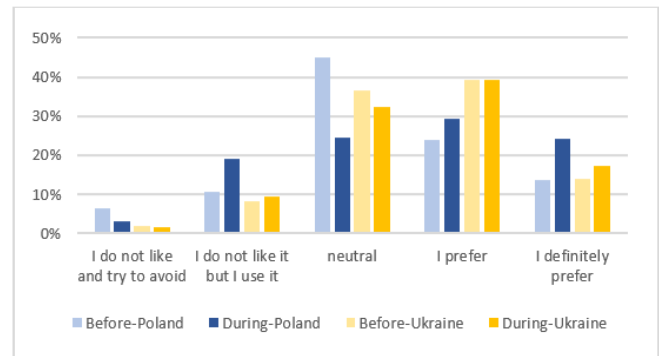


Fig. 1. Respondents' attitudes towards online communication

contacts, but a visible increase in their intensity.

In all categories, the changes in attitudes towards online communication were slightly positive with a greater difference among females (Table V). In both Poland and Ukraine, we saw a difference according to the age: the scale of positive attitude change in the Older group is much higher than the average. This means that a significant part of this group - forced to use online communication due to the circumstances of the pandemic - gained a more positive opinion of it, while the Young and Medium groups hardly changed their opinion (perhaps because they were already used to this form).

RQ5: Attitudes towards Online Education. Both Polish and Ukrainian men preferred distance learning more than women (before and during the pandemic), but these preferences were stronger among Ukrainian men. However, as Table VI shows, women changed their preferences more than men, in favor of distance learning. Similarly to changes in online communication, we assessed the changes in the attitudes towards online

TABLE V
CHANGES IN ATTITUDES TOWARDS ONLINE COMMUNICATION

		Poland	Ukraine
Average		0.16	0.06
Gender	Female	0.17	0.08
	Male	0.13	0.06
Age	Young	0.05	0.05
	Medium	0.15	0.07
	Older	0.30	0.27

TABLE VI
RESPONDENTS' DISTRIBUTION IN ACCORDANCE TO ATTITUDES'
CHANGES TOWARDS DISTANCE LEARNING

			I do not like	Neutral	I prefer
Total	Poland		5%	-20%	15%
	Ukraine		1%	-10%	8%
Gender	Poland	Female	5%	-23%	17%
		Male	-12%	-16%	13%
	Ukraine	Female	-1%	-12%	12%
		Male	4%	-8%	6%
Age	Poland	Young	-1%	-14%	14%
		Medium	7%	-23%	16%
		Older	3%	-19%	16%
	Ukraine	Young	2%	-9%	7%
		Medium	0%	-7%	7%
		Older	7%	-40%	33%

education. A positive change was visible in both countries (0.25 for Poland and 0.13 for Ukraine, respectively), but greater in Poland. Taking into account the gender criterion, the positive change in attitude was greater in the case of women (especially in the case of Ukraine where it was over three times higher than men). In the Young and Medium groups the scale of changes was similar to the average for both countries, while in the Older group it was greater than average: in the case of Ukraine it is over three times higher than the average.

V. DISCUSSION

The differences between digital competences of Polish and Ukrainian respondents might have three fold explanation: (1) Ukrainian youth enter higher education on average 2 years earlier than their Polish counterparts, which is a huge difference at this age; (2) differences in the level of IT education at the earlier stages, and (3) economic differences between Poland and Ukraine, resulting in lower availability of computer equipment in Ukraine. Interestingly, a much greater share of Ukrainian women than men assessed their IT skills as below average. It seems that Ukrainians are aware of some deficiencies in digital competences, and that is why actions have been taken to develop these skills by students [12], [13].

In general, Ukrainians perceive their technical conditions as worse than Poles. In Ukraine, a lot of students come from villages, without access to the Internet. Ukrainian families on average are much poorer than Polish ones; sometimes Ukrainian students do not have private computers and they can only use computers at universities, having access to the Internet via their mobile phones. Technical problems mainly concern younger Ukrainian respondents, which might be explained by their strong economic dependence on parents (for whom the Internet does not have to be a priority, especially in a generally poor economic condition).

Similarly to the findings in [4] we noticed the deterioration in students' wellbeing. However, contrary to the research

results reported by [8] we did not notice considerable differences in wellbeing deterioration between Polish and Ukrainian respondents: indeed slightly more visible wellbeing deterioration was detected among Polish respondents. At the beginning of the pandemic there were severe restrictions on face-to-face communication, which may be the reason for the deterioration of the wellbeing of respondents. It should be noted that in Ukraine the survey was conducted at the time when learning at universities was on premise (contrary to Poland), hence Ukrainian respondents' memory might fade as far as bad distance learning experience is concerned. Improvement of the wellbeing of the Polish Older group of respondents might be explained by the fact that this group contains students above the age of 24, including part-time students, for whom remote learning is much more attractive due to less organizational effort and resources required to study in this mode.

Some Ukrainian students have limited access to computers and that is why the pandemic has not caused such a great increase in the number of hours spent online like in the case of Polish respondents. Besides, a lot of Ukrainian students take part in distance courses offered by various companies that also increase the total number of hours online. Not surprisingly, learning activities contributed the most to the increase in time spent online during the pandemic. In general, before the pandemic, classes in HEIs in Ukraine (with a few exceptions) were conducted on premise, in connection with which the transition to distance learning caused a sharp increase in the number of hours that Ukrainian students spent online for educational purposes.

There were significant differences in online behaviour patterns between Polish and Ukrainian students: in Ukraine there was a slight decrease in personal contacts and a high increase in online contacts; in Poland there was a huge decrease in face-to-face contacts and a small decrease in online contacts. The decrease in online contacts in Poland might be the result of the decrease in face-to-face contacts – when restrictions were severe and the number of face-to-face contacts decreased, then online contacts resulting from them also decreased. Pandemic regulations were similar, hence the differences between the results might be attributed to culture differences. In Ukraine a lot of families were forced by the pandemic to buy devices that facilitate online communication, and that is why its increase is noticeable.

Interestingly, during the pandemic almost the same numbers of face-to-face contacts were reported in both countries – 8 in Poland and 9 in Ukraine. Maybe this convergence defines the social sphere, but this topic would require further investigation. Differences in the frequency of online communication might be attributed to the differences in device ownership: Polish students usually have computers with Internet access that allow them to communicate constantly or several times a day, whereas some Ukrainian students only use computers with Internet access at campuses, which allows them to communicate online once a day or a few times a week. There is a difference in the behaviour between Poles and Ukrainians: in Poland the decreases in face-to-face contacts

are greater, however it seems that older students in Poland transferred some contacts from face-to-face to online. In the case of Ukraine such an interpretation may apply to all age groups. This might be caused by the increased availability of communication equipment in Ukraine.

In Poland most respondents were neutral towards online communication. This might be explained by the fact that they have had access to this way of communication for a long time and might be tired of it being a part of everyday life. On the contrary, in Ukraine, respondents seem to like this way of communication. Maybe they still treat it as something new that opens up more possibilities.

The quality of distance learning in Ukraine might be lower than in Poland since this country has less experience in this type of teaching, however at the time of conducting the survey students had classes on premise and that is why they might be more in favour of distance learning. In Poland the usage of distance learning was more intense and some students might be fed up with it. Such a recurring difference between the members of the Older group and the other groups may result from the fact that earlier these students had no habit, or even had resistance, to online learning, and using this form, being forced by the pandemic, could significantly change this attitude. Interestingly, women change their attitude towards online learning to much more positive than men, similarly to older students.

In further research, we would like to investigate thoroughly different aspects of online learning, focusing mainly on the perspectives of online learning and the model of education after the pandemic.

APPENDIX

Survey items related to the current study

- 1) Gender; Age
- 2) Digital competences: beginner; below average; average; above average; professional
- 3) Technical conditions (ICT): constant problems; frequent problems; sufficient for basic needs; occasional problems; have no problems
- 4) The number of hours spent online per week (not counting professional work, but including studies) is approximately (up to 15h; 16-25h; 26-35h; 36-45h; over 45h)
- 5) The number of hours per week related to online studies
- 6) Estimated number of friends in regular socializing via electronic media (social media)
- 7) Estimated number of friends in face-to-face contacts
- 8) The frequency of communicating with friends via electronic media (occasionally; a few times a week; once a day; several times a day; constantly)
- 9) Attitude towards Internet communication
- 10) Attitude towards distance learning (e.g. training videos on Youtube)

- 11) Mental health change after switching to distance learning (significantly deteriorated; worsened; no change; improved; significantly improved)

ACKNOWLEDGMENT

The publication has been financed by the subsidy granted to the Cracow University of Economics - Project no. 032/SD/2022/PRO.

REFERENCES

- [1] I. Bondar, A. Humenchuk, Y. Horban, L. Honchar, and O. Koshelieva, "Conceptual and innovative approaches of higher education institutions (HEIs) to the model of training a successful specialist formation during a covid pandemic," *Journal of Management Information and Decision Sciences*, vol. 24, 2021, pp. 1–8.
- [2] M. R. D.Center, "Learning from student browsing data on E-learning platforms: case study," *In Position Papers of the 2020 Federated Conference on Computer Science and Information Systems*, 2020, pp. 37. DOI: 10.15439/2020F138
- [3] R. Afroz, N. Islam, S. Rahman, and N. Z. Anny, "Students' and teachers' attitude towards online classes during COVID-19 pandemic: a study on three Bangladeshi government colleges," *Research in Business & Social Science*, vol. 10, 2021, pp. 462–476. DOI: <https://doi.org/10.20525/ijrbs.v10i3.1155>
- [4] Z. Kawczyńska-Butrym, V. Pantyley, M. Butrym, G. Kisla, and L. Fakeyeva, "Students in times of pandemic: employment, living conditions, and health. Case Studies from Poland, Ukraine, and Belarus," *Geographia Polonica*, vol. 94, 2021, pp. 429–440. DOI: <https://doi.org/10.7163/GPol.0213>
- [5] P. Tsatsou, "Digital inclusion of people with disabilities: a qualitative study of intra-disability diversity in the digital realm," *Behaviour and Information Technology*, vol.39, 2020, pp. 995–1010. DOI: <https://doi.org/10.1080/0144929X.2019.1636136>
- [6] V. Belan, "Using distance learning technologies for training future teachers of professional technical courses at the universities of the Republic of Poland and Ukraine," *Professional Pedagogic*, vol. 2, 2020, pp. 145–152. DOI: <https://doi.org/10.32835/2707-3092.2020.21.145-152>
- [7] Y. Klapkiv and T. Dluhopolska, "Changes in the tertiary education system in pandemic times: Comparison of Ukrainian and Polish universities," *Revista Romaneasca pentru Educatie Multidimensionala*, vol. 12, 2020, pp. 86–91. DOI: <https://doi.org/10.18662/rrem/12.1sup2/250>
- [8] P. Długosz and L. Kryvachuk, "Neurotic generation of Covid-19 in Eastern Europe," *Frontiers in Psychiatry*, vol. 12, 2021, pp. 1–8. DOI: <https://doi.org/10.3389/fpsy.2021.654590>
- [9] D. Ochnik, A. M. Rogowska, C. Kuśnierz, M. Jakubiak, A. Schütz, M. J. Held, A. Arzenšek, J. Benatov, R. Berger, E. V. Korchagina, I. Pavlova, I. Blažková, I. Aslan, O. Çınar, and Y. A. Cuero-Acosta, "Mental health prevalence and predictors among university students in nine countries during the COVID-19 pandemic: a Cross-national study," *Scientific Reports*, vol. 11, 2021, pp. 1–13.
- [10] D. Dymek, M. Grabowski, and G. Paliwoda-Pękosz, "Change of students' online behavioural patterns during the COVID-19 pandemic: insights from Poland," in *Proceedings of the International Business Information Management Association Conference*, vol. 38, 2021, pp. 3263–3272.
- [11] S. Didkivska and T. A. Vakaliuk, "Insights from Ukrainian students on distance learning During the Covid-19 Pandemic," in *Proceedings of the International Business Information Management Association Conference*, vol. 38, 2021, pp. 1569–1578
- [12] T. A. Vakaliuk, V. Kontsedailo, D. Antoniuk, V. Korotun, S. Semerikov, and I. Mintii, "Using game Dev Tycoon to develop professional soft competencies for future engineers-programmers," in *CEUR Workshop Proceedings*, 2020. DOI: <http://dx.doi.org/10.2139/ssrn.3719840>
- [13] B. Kovalchuk and A. Zaika, "Formation of digital competence of future masters of industrial training of agricultural profile," *Information Technologies and Learning Tools*, vol. 85, 2021, pp. 118–129. DOI: <https://doi.org/10.33407/itlt.v85i5.3897>

A Look at Evolution of Teams, Society, Smart Cities, and Information Systems based on Patterns of Primary, Adaptable, Information, and Creative Society

Dmitriy Gakh

Institute of Control Systems of ANAS
Bakhtiyar Vahabzadeh str. 68, Baku, Azerbaijan
Email: dmgakh@gmail.com
ORCID: 0000-0002-3007-8891

Abstract—Development of information and communication technologies (ICTs) and society are interdependent. Smart City (SC) is one conception emerging as a result of such interdependence. This paper considers evolution of teams, society, SCs, and Information Systems based on Patterns of Primary, Adaptable, Information, and Creative Society. The evolution is described in 16 levels according to the Simple Learning Motivations Hierarchy Model (SLMHM). Success factors of ICT projects and strategy for SC development are discussed from point of view of patterns considered.

I. INTRODUCTION

THERE is a gap between development levels of society and technologies. Such phenomena are observed during globalization processes where developed countries try to adopt technologies in undeveloped ones. Artificial Intelligence (AI) threat to humanity is an example of a gap between levels of society development and ICTs. Another gap relates to 5G cellular networks, where some researchers cannot say if this technology is dangerous, or not [1, 2].

Structure of society patterns, including Initial Formation (IFP), Primary (PSP), Adaptable (ASP), Information (ISP), and Creative (CSP) (the five patterns in further text)[3], is considered. The five patterns are based on idea of four organization cultures, i.e. Impulsive, Dependent, Independent, Interdependent [3, 4]. Development of society and ICT solutions are considered according to 16 levels of the SLMHM model (see [5] and Fig 1 for details). The discussion includes conception of SC [6-12] and several maturity models [8, 10, 12-20]. Understanding of demand in information technologies and information systems (IT/IS) according to the five patterns and SLMHM levels is a success factor of ICT projects.

II. METHODOLOGY, THEORETICAL BACKGROUND, AND RESEARCH QUESTIONS

This research is theoretical and based on observations and literature analyses. The methodology is based on the 1) literature analyses; 2) juxtaposition of the five patterns, SLMHM levels, and IT/IS (see Fig. 1); 3) discussion of findings and their possible practical application.

Ecological, economic, socio-cultural, and political components were proposed to be considered as ones of the Sustainable Information Society (SIS) [21]. Technological, economic, occupational, spatial, and cultural elements of the Information Society were selected [22]. 17 UN sustainable development goals [23] can be achieved by the harmonization of society and the environment. ICTs provide valuable tools for such harmonization allowing intensive information processing and control [3,5]. Rabie describes human society development ages as the following: hunting and gathering, agricultural, industrial, knowledge (post-industrial, the information, the globalization) [24].

Cities worldwide play a prime role in social and economic aspects and have a huge impact on the environment [6]. Although the meaning of SC is not settled yet, there is an agreement on the significant role of ICTs in smart urban development [7]. ICTs are an important tool for the transformation of industrial society into information and knowledge society. It is a networked society, emerging a new social morphology, and gaining economic, social, political, and cultural primacy. Society is constantly changing, either because the economic, social, political, and cultural contexts are increasingly massified, internationalized, and globalized, or because the relations of life, study, work, and capital are changing rapidly and constantly [25].

There are also several mature models allowing to assess how “smart” the city is. The low level of maturity of the SC, the lack of sufficient and real-time data, and the lack of standardization in previous years hampered the development of such models. Citizens are considered “prosumers” of geo-tagged data and content affecting cities’ everyday norms and interactions [8]. The study shows that SC can be considered a socio-technical system [9].

SC Development Maturity Model, defined in [10] consists of five levels: initial, repeated, defined, managed, and optimized, and is built based on six factors for which the maturity is determined on these five levels which are strategic alignment, culture, people, governance, method, and IT/IS.

Software Capability Maturity Model (SW CMM) was developed by the Software Engineering Institute (SEI) at Carnegie Mellon University in 80s-90s as a response to a re-

quest to provide the government with a method for assessing the capability of their software contractors [13]. It should be mentioned, that SW CMM was developed for organizations, but not for country-level projects [14, 15]. Since the introduction of the Capability Maturity Model, many maturity models were developed including Knowledge Management Maturity Model (KMMM) [16].

There are ISO 37xxx standards relating to establishment of SC operating models for sustainable communities [26-28]. There are ranking SC models and experience of ranking cities [11, 29]. The SC Maturity and Benchmark Model have been designed to capture the key aspects of a city's transformation journey to a smarter city. The model allows a city to quickly assess its strengths and weaknesses in five key dimension areas related to city smartness and set clear goals as to how it wishes to transform over the next two-five years [12]. Three maturity models, which approached a city in a holistic way were discussed in [17, 18]. Main maturity model design patterns were identified. Typically, three application-specific purposes of the model used, i.e. descriptive, prescriptive, and comparative are distinguished [19]. Different Maturity Models for Information Systems were studied. The number of levels in the considered models varies from 4 to 6 [20].

Sarkar describes quadri-dimension economy as a part of Progressive Utilization Theory (PROUT) [30]. These 4 parts of economy intuitively match 4 McWinney's realities [31, 32]. Several economic conceptions were taken from Life Cycle Management [33-35] and from the experience economy [36]. Types of dystopia that are described in [37] intuitively match to the considered patterns.

The following research questions are:

RQ1. What are the stages of society development from ICTs point of view?

RQ2. How do stages and trends of society development relate to ICTs and how this relation can be used for ICT projects?

III. THE MODEL

Fig. 1 outlines the five patterns, SLMHM levels, and related IT/IS. The five patterns are similar to the development stages. Main features of the model include:

- society develops consequentially from Initial Formation to Primary, then to Adaptable, then to Information, and finally to Creativity stages;
- SLMHM allows modeling the team/society development by more detailed steps (levels);
- transition to specific level requires satisfaction of requirements of all previous levels;
- there are no clear boundaries between development stages;
- the structure of society contains IFP, PSP, ASP, ISP, and CSP in different proportions;
- the prevailing pattern shows which technologies are demanded;

- applicability to groups of any size, from small teams up to large organizations and society;
- can be integrated with other models vertically (by relevant layers) and horizontally (by breaking up the area of interest into parallel developing parts).

IV. DISCUSSION OF THE FINDINGS

There is intuitive match between the five patterns and different economical (four Whitmore's cultures [4], dimensions of sustainability [33], etc.) and philosophical (PROUT [30], McWinney's realities [31, 32], etc.) conceptions. The match is discussed in detail in [3]. According to [28], the model, presented in Fig. 1 corresponds to all three levels of SC standards: Strategic, Process, and Technical Specifications. There is no direct accordance between the CMM [10] and SLMHM levels. Meanwhile, one can say that some match between CMM and SLMHM levels exists.

Part of the IFP in society can be assessed through a percentage of the informal economy and unemployment rate. So, the contribution of the informal sector, excluding agriculture, to GDP in developing countries is about 16.3% (Venezuela, 2006) up to 61.8% (Benin, 2000) [38]. The study presented in [39] shows higher figures for informal non-agricultural employment – from 46.2% (Hanoi) up to 83.1% (Lomé). One can say that exclusion of the agricultural sector from considering informal employment reflects the fact that the agricultural sector relates to the agricultural age according to Rabie [24], i.e. to the PSP. Entertainment ICT solutions seems the only ICT solutions demanded in this pattern.

PSP represents the structure (skeleton) of the society. Its part can be estimated by level of bureaucracy and traditions. Here database solutions seems the most demanded ones. ASP is based on logic and facts. Industry and production are the main components of the ASP, while analytics and optimization are the main components of ICTs. ISP is characterized by connectivity and networks. The main values of this pattern relate to the human feelings. That's why social media and networks have become so popular nowadays. CSP is the next pattern, characterized by the creativity, ideas, and expansion. The key technologies here are Artificial Neural Networks (ANNs), AI, and blockchain. Although mankind is in the information age (ISP), the entry into the post-information age (age of creativity / CSP) is characterized by the development of these technologies. This is possible because there are no clear boundaries between development stages.

IFP, PSP, and ASP relate to reactive approaches to the changes. IFP is not considered in many cases because it does not relate to a team or complete society. The main aim of ASP is to achieve adaptation to changes and sustainability. ISP and CSP relate to proactive approaches [41] to the changes and sustainable development.

There is a need for a maturity model for maturity modeling [42]. SLMHM can be used to design other development and maturity models.

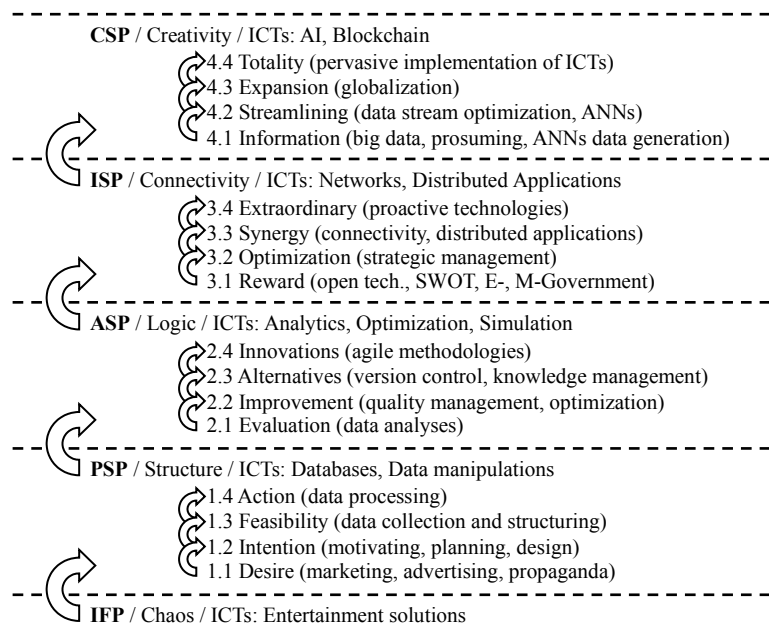


Fig 1. The Five Patterns, SLMHM Levels, and IT/IS

Examples of problems that can be explained by the considered model are presented in Fig. 1. They include:

- AI threat problem. AI relates to CSP. CSP requires developed ISP. ISP is currently developing (information age). It means that developed connectivity is a key to solve such problems;
- teams that have not adopted quality management cannot gain full benefits from technologies relating to SLMHM level 2.3 and higher;
- sustainability can be achieved only if SLMHM level 2.4 is achieved. Sustainable development can be achieved only if SLMHM level 3.4 is achieved;
- stage of society development can be assessed in two ways. 1) by studying the culture; 2) by studying of effectiveness of implemented ICT solutions;
- implementation of higher level technologies in societies with lower level is risky. Assurance that society level is the same or higher than that of implemented IT/IS is a recipe for success;
- implementation of technologies of higher level before appropriate implementation of technologies of lower levels is risky;
- moral decision making by algorithms [43] is not a trivial problem because modern algorithms relate to ASP. At the same time, moral relates to human feelings, in other words, to ISP. The development of ISP is the key to solving the problem.

There are strong reasons to assert that the implementation of ICT solutions in societies with lower level will contribute to raising the society's level, because society and penetrated ICT solutions form one system.

V. CONCLUSION

The five patterns corresponding to the development stages, 16 SLMHM levels, and corresponding ICTs were considered (see Fig. 1). Match of society development level and relevant ICTs is considered as a success factor of ICT projects implementation. The ICT projects are essential in realization of SC conception.

Disadvantages of this research include small scope of this paper, considering ICTs briefly, and theoretical nature of the research. Practical implementation of considered theory requires methodology for assessing levels of society development and high volume of computations [5].

Future research should include detailing of Fig. 1 by adding ICTs and development of society assessment methodology. Practical implementation of the theory has the special interest.

REFERENCES

- [1] M. Karaboytcheva, "Effects of 5G wireless communication on human health", European Parliamentary Research Service, 2020.
- [2] H. Hasan, A. Yosef, H. Hachem, "5G Radiation and Potential Risks to the Environment and Human Health", 2021, Turkish Journal of Computer and Mathematics Education, vol.12 No.6, pp. 1689-1693. <https://dx.doi.org/10.17762/turcomat.v12i6.3376>
- [3] D. Gakh, "A Look to Model of Society and Teams Development based on Initial Formation, Primary, Adaptable, Information, and Creative Society Patterns". Preprints 2022, 2022080473. <http://dx.doi.org/10.20944/preprints202208.0473.v1>
- [4] J. Whitmore, "Coaching for Performance", 5th ed., Nicholas Brealey Publishing, 2017.
- [5] D. Gakh, "Education Development Strategy on Base of the Analysis of Messages in the Russian-Speaking Segment of the Internet using SLMHM Model", 2022. <http://dx.doi.org/10.35542/osf.io/cnh6q>
- [6] V. Albino, U. Berardi, R. Maria Dangelico, "Smart Cities: Definitions, Dimensions, Performance, and Initiatives", Journal of Urban

- Technology, 2015, vol. 22, no. 1, pp. 3–21. <https://dx.doi.org/10.1080/10630732.2014.942092>
- [7] F. Mosannenzadeh, D. Vettorato, “Defining Smart City. A Conceptual Framework Based on Keyword Analysis”, *TeMA*, 2014, <http://dx.doi.org/10.6092/1970-9870/2523>
- [8] V. Moustaka, A. Maitis, A. Vakali, L. Anthopoulos, “CityDNA Dynamics: A Model for Smart City Maturity and Performance Benchmarking”, *WWW '20: Companion Proc. of the Web Conf.*, 2020, pp. 829–833, <http://dx.doi.org/10.1145/3366424.3386584>
- [9] H. Kopackova, P. Libalova, “Smart city concept as socio-technical system”, 2017 International Conference on Information and Digital Technologies (IDT), Zilina, 2017, pp. 198-205, <http://dx.doi.org/10.1109/DT.2017.8024297>
- [10] S. Waarts, “Smart City Development Maturity. A study on how Dutch municipalities innovate with information using a smart city development maturity model”, Master Thesis. Tilburg University, 2016.
- [11] A. Aihemaiti (E. Ahmet), A. ZAİM, “Ranking Model of Smart Cities in Turkey”, *Anatolian Journal of Computer Sciences*, 2018, vol:3 no:2, pp: 35-43.
- [12] TM Forum, “Smart City Maturity & Benchmark Model”, retrieved Jul 19, 2022 from <https://www.tmforum.org/smart-city-forum/smart-city-maturity-benchmark-model>
- [13] M. Paulk, “A history of the Capability Maturity Model for Software”, 2001, retrieved Aug 23, 2022 from <https://citeseerx.ist.psu.edu/viewdoc/download?rep=rep1&type=pdf&doi=10.1.1.216.199>
- [14] Q. Pham, “Measuring the ICT maturity of SMEs”, *Journal of Knowledge Management Practice*, 2010, vol. 11, no.1, pp. 34–40, retrieved Jul 19, 2022 from <https://bit.ly/3C7KyJ7>
- [15] X. Pham, N. Le, T. Nguyen, “Measuring the ICT maturity of enterprises under uncertainty using group fuzzy ANP”, *Int J Mach Learning Comput*, 3 (6), 2013, pp. 524–528, <http://dx.doi.org/10.7763/IJMLC.2013.V3.374>
- [16] N. Qodarsih, Handayani, R. Sabtiana, “Knowledge Management Maturity Model: A Case Study at Ministry XYZ”, *Advances in Intelligent Systems Research*, volume 172, Sriwijaya International Conference on Information Technology and Its Applications (SICONIAN 2019), <http://dx.doi.org/10.2991/aisr.k.200424.026>
- [17] H. Shoukry, “To what extent is your city smart? – Smart Cities Maturity Models”, *Zigurat Global Institute of Technology*, 2021, retrieved Jul 19, 2022 from <https://www.e-zigurat.com/blog/en/smart-cities-maturity-models>
- [18] P. Torrinha, R. Machado, “Assessment of maturity models for smart cities supported by maturity model design principles”, 2017 IEEE International Conference on Smart Grid and Smart Cities (ICSGSC), <http://dx.doi.org/10.1109/ICSGSC.2017.8038586>
- [19] J. Poeppelbuss, M. Roeglinger, “What makes a useful maturity model? A framework of general design principles for maturity models and its demonstration in business process management”, 2011, Conference: European Conference on Information Systems (ECIS) At: Helsinki, Finland Volume: 19.
- [20] D. Proença, J. Borbinha, “Maturity Models for Information Systems - A State of the Art”, Conference on ENTERprise Information Systems / International Conference on Project Management / Conference on Health and Social Care Information Systems and Technologies, CENTERIS / ProjMAN / HCist 2016, October 5-7, 2016. <http://dx.doi.org/10.1016/j.procs.2016.09.279>
- [21] E. in Ziembra, “The ICT adoption in enterprises in the context of the sustainableformation society”. In: Proceedings of the Federated Conference on Computer Science and Information Systems, 2017, vol. 11, pp. 1031–1038. <https://dx.doi.org/10.15439/2017F89>
- [22] F. Webster, “Theories of the Information Society”, 4ed, Routledge, 2014. ISBN 9780415718790
- [23] United Nations, “17 Goals of Sustainable Development”, retrieved Jul 19, 2022 from <https://sdgs.un.org/goals>
- [24] M. Rabie, “The Development of Human Societies”, *Saving Capitalism and Democracy*, Palgrave Macmillan, New York, 2013. https://dx.doi.org/10.1057/9781137321312_3
- [25] J. Rascão, N. Poças, “Freedom of Expression and the Right to Privacy and Ethics in Dialectic of Human Rights in This Complex and Turbulent Society”, *IJMPMA* vol.9, no.2, pp.1-28, 2021. <http://dx.doi.org/10.4018/IJMPMA.2021070101>
- [26] BSI, “Sustainable cities and communities: ISO 37106. Summary guidance on establishing smart city operating models for sustainable communities”.
- [27] “ISO/TS 37151:2015 Smart community infrastructures - Principles and requirements for performance metrics”, Standardization, I. the I. O. for. 2015, retrieved Jul 19, 2022 from http://www.iso.org/iso/catalogue_detail?csnumber=61057
- [28] BSI, “Making cities smarter. Guide for city leaders: Summary of PD 8100”, Department for Business Innovation & Skills.
- [29] R. Giffinger, C. Fertner, H. Kramar, R. Kalasek, N. Pichler-Milanovi c, E. Meijers, “Smart cities: Ranking of european medium-sized cities”, Vienna UT: Centre of Regional Science, 2007, retrieved Jul 19, 2022 from http://www.smart-cities.eu/download/smart_cities_final_report.pdf.
- [30] P. Sarkar, “Quadri-Dimensional Economy”, “Prout in a Nutshell”, vol 3. , Ananda Marga Publications, Calcutta 2020.
- [31] W. McWhinney, “Growing Into the Canopy”, *Journal of Transformative Education*, vol. 5 no. 3 pp. 206-220, 2007. <https://doi.org/10.1177/1541344607307023>.
- [32] W. McWhinney, J. Webber, D. Smith, B. Novokowsky, “Creating Paths of Change: Managing Issues and Resolving Problems in Organizations”, SAGE Publications, 1997.
- [33] G. Sonnemann, E. Gemechu, A. Remmen, J. Frydendal, A. Jensen (2015), “Life Cycle Management: Implementing Sustainability in Business Practice”, In: Sonnemann, G., Margni, M. (eds) *Life Cycle Management. LCA Compendium – The Complete World of Life Cycle Assessment*. Springer, Dordrecht. https://doi.org/10.1007/978-94-017-7221-1_2
- [34] E. Gemechu, G. Sonnemann, A. Remmen, J. Frydendal, A. Jensen (2015), “How to Implement Life Cycle Management in Business?”, In: Sonnemann, G., Margni, M. (eds) *Life Cycle Management. LCA Compendium – The Complete World of Life Cycle Assessment*. Springer, Dordrecht. https://doi.org/10.1007/978-94-017-7221-1_4
- [35] M. Baitz, (2015). “From Projects to Processes to Implement Life Cycle Management in Business”, In: Sonnemann, G., Margni, M. (eds) *Life Cycle Management. LCA Compendium – The Complete World of Life Cycle Assessment*. Springer, Dordrecht. https://doi.org/10.1007/978-94-017-7221-1_8
- [36] B. Pine II, J. Gilmore, 2020, “The experience economy: Competing for customer time, attention, and money”, Harvard Business School Press, Boston.
- [37] Read Write Think, “Dystopias: Definition and Characteristics” (PDF). Archived (PDF) from the original on 23 Sept. 2010, retrieved Jul 19, 2022 from http://www.readwritethink.org/files/resources/lesson_images/lesson926/DefinitionCharacteristics.pdf
- [38] United Nations, “Enhancing Productivity in the Urban Informal Economy”, United Nations Human Settlements Programme (UN-Habitat), 2016.
- [39] D. Brown, G. McGranahan, “The urban informal economy, local inclusion and achieving a global green transformation”, *Habitat International*, April 2016, vol. 53 pp. 97-105. <https://dx.doi.org/10.1016/j.habitatint.2015.11.002>
- [40] D. Billsus, D. Hilbert, D. Maynes-Aminzade, “Improving Proactive Information Systems”, *IUI '05: Proceedings of the 10th international conference on Intelligent user interfaces*, 2005, pp. 159–166. <https://dx.doi.org/10.1145/1040830.1040869>
- [41] H. Holz, H. Maus, A. Bernardi, O. Rostanin, “From Lightweight, Proactive Information Delivery to Business Process-Oriented Knowledge Management”, *Journal of Universal Knowledge Management*, 2005, no. 2, pp. 101-127.
- [42] J. van Hillegersberg. “The Need for a Maturity Model for Maturity Modeling”. In: Bergener, K., Räckers, M., Stein, A. (eds) *The Art of Structuring* pp 145–151. Springer, Cham. 2019. https://dx.doi.org/10.1007/978-3-030-06234-7_14
- [43] G. Miller. “Artificial Intelligence Project Success Factors—Beyond the Ethical Principles”. In: Ziembra, E., Chmielarz, W. (eds) *Information Technology for Management: Business and Social Issues. FedCSIS-AIST ISM 2021* 2021. Lecture Notes in Business Information Processing, vol 442. Springer, Cham. https://doi.org/10.1007/978-3-030-98997-2_4

Identifying Reliable Sources of Information about Companies in Multilingual Wikipedia

Włodzimierz Lewoniewski, Krzysztof Węcel, Witold Abramowicz

Department of Information Systems, Poznan University of Economics and Business

Al. Niepodległości 10, Poznan 61-875, Poland

Email: {wlodzimierz.lewoniewski, krzysztof.wecel, witold.abramowicz}@ue.poznan.pl

Abstract—For over 21 years Wikipedia has been edited by volunteers from all over the world. Such editors have different education, cultural background and competences. One of the core rules of Wikipedia says, that information in its articles should be based on reliable sources and Wikipedia readers must be able to verify particular facts in text. However, reliability is a subjective concept and a reputation of the same source can be assessed differently depending on a person (or group of persons), language and topic. So each language version of Wikipedia may have own rules or criteria on how the website must be assessed before it can be used as a source in references. At the same time, nowadays there are over 1 billion websites on the Internet and only few developed Wikipedia language versions contain non-exhaustive lists of popular websites with reliability assessment. Additionally, since reputation of the source can be changed during the time, such lists must be updated regularly.

This study presents the result of identification of reliable sources of information based on the analysis of over 200 million references that were extracted from over 40 million Wikipedia articles. Using DBpedia and Wikidata we identified articles related to various kinds of companies and found the most important sources of information in this area. This also allows to compare differences of the source reliability between Wikipedia languages.

I. INTRODUCTION

INFORMATION presented in Wikipedia articles should be based on reliable sources [1]. The source can be understood as the work (book, paper etc.), author, publisher. Such sources must have a proper reputation, should present all majority and significant minority views on some piece of information. Following this rule ensures that readers of the article can be assured that each provided specific fact (piece of information or statement) comes from a published and reliable source. Hence, before adding any information (even if it is a generally accepted truth) to this online encyclopedia, Wikipedia volunteer editors (authors or users) need to ascertain whether the facts put forward in the article can be verified by other people, who read Wikipedia [2].

Few developed language versions of Wikipedia contain non-exhaustive list of sources whose reliability and use on Wikipedia are frequently discussed. Even the English Wikipedia (the largest chapter of the encyclopedia) has such general list with information on reliability for less than 400 websites [3]. Sometimes we can find such lists for specific topics (e.g. video games, films, new Wikipedia articles in English Wikipedia).

It could take a significant human effort to produce a more complete list of assessed internet sources - there are over

billion websites available in the Internet [4], [5] and a lot of them can be considered as a source of information. So, it can be very challenging and time consuming task for Wikipedia volunteers to assess reliability of each source. Moreover, reputation of each website can change with time - hence, such lists must be updated regularly. Additional challenge - each source may have a different reliability score depending on topic and language version of Wikipedia.

More complete and updated list of reliable sources can be useful not only for Wikipedia editors, but also for readers of this popular encyclopedia. The aim of this study is to show some possibilities of automating this process by analyzing existing and accepted content with sources in Wikipedia articles about companies in different languages. This paper uses existing and new models for reliability and popularity assessment of websites. The results show that depending on models it is possible to find such important sources in selected Wikipedia languages. Additionally, we show how the assessment of same sources can vary depending on language of this encyclopedia.

II. RELATED WORKS

Researching the quality of Wikipedia content is a fairly developed topic in scientific works. As one of the key factors influencing the quality of Wikipedia articles is the presence of references, some studies focused on researching information sources. Some of works use the number of references to automatically assess quality of the information in Wikipedia [6], [7], [8]. Such important measures are implemented in different approaches for automatic quality assessment of Wikipedia articles (for example WikiRank [9]). References often contain external links (URL addresses) where cited information is placed. Such links in references can be assessed by indicating the degree to which these conform to their intended purpose [10]. Moreover, those links can be employed separately to assess quality of Wikipedia articles [11], [12].

Some of the studies focused on metadata analysis of the sources in Wikipedia references. One of the previous works used ISBN and DOI identifiers to unify the references and find the similarity of sources between various Wikipedia language editions [13]. It is increasingly common practice to include scientific sources in references of Wikipedia articles. [13], [14], [15], [16]. At the same time, it is worth noting that such references often link to open-access works [17] and recently published journal articles [18]. One of the studies devoted

to the COVID-19-related scientific works cited in Wikipedia articles and found that information comes from about 2% of the scientific works published at that time [19].

News websites are also one of the most popular sources of the information in Wikipedia and there is a method for automatic suggestion of the news references for the selected piece of information [20]. Particularly popular are references about recent content or life events [21]. For example in case of information related to COVID-19 pandemic Wikipedia editors inclined to cite the latest scientific works and insert more recent information on to Wikipedia shortly after the publication of these works [19].

Previous relevant publication [15] to this paper proposed and implemented 10 models for sources evaluation in Wikipedia articles. Results of assessment are also implemented in online tool "BestRef" [22]. Such approaches uses features (or measures) that can be extracted from publicly available data (Wikimedia Downloads [23]), so anybody can use those models for different purposes. One of the recent studies [24] in addition to the proposed models included also a time dimension to show how importance of the given web source of information on COVID-19 pandemic can be changed over different months.

III. REFERENCES EXTRACTION

To be able to extract information about references we prepared own parser in Python and applied it on Wikimedia dumps with articles in HTML format [23]. Table I presents general statistics of the extraction.

External links (or URL addressees) in references were used to indicate main address of the website. However, each web source can use different structure of URL addresses. For example, some of the websites can use subdomains for separate topics of information or news. Another example - some organizational units (e.g. departments) of the same company may post its information on separate subdomains of main organization. To detect which level of domain indicates the source this work uses the Public Suffix List, which is a cross-vendor initiative to provide an accurate list of domain name suffixes [25]. Figure 1 presents example of URL address at fourth level domain with indication of main website.

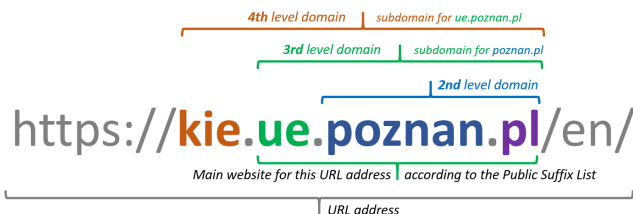


Fig. 1. Example of URL address at fourth level domain with indication of main organizational website using the Public Suffix List

Reference per Article (RpA) value shows average number of references in Wikipedia articles (in case of table I among

TABLE I
STATISTICS ON REFERENCES EXTRACTION FROM WIKIPEDIA ARTICLES IN DIFFERENT LANGUAGES. SOURCE: OWN CALCULATIONS BASED ON WIKIMEDIA DUMPS IN APRIL 2022.

Abbr.	Language	Articles	References	Uniq. refs	RpA
ar	Arabic	1,162,992	6,689,241	5,208,058	5.75
be	Belarusian	216,747	589,402	453,54	2.72
bg	Bulgarian	280,546	935,65	727,127	3.34
ca	Catalan	698,608	3,350,195	2,637,219	4.80
cs	Czech	500,923	2,358,219	1,711,325	4.71
da	Danish	274,091	765,275	616,9	2.79
de	German	2,678,208	12,737,779	10,110,149	4.76
el	Greek	208,442	1,644,945	1,295,992	7.89
en	English	6,477,118	70,355,363	52,040,192	10.86
eo	Esperanto	315,637	302,146	257,393	0.96
es	Spanish	1,764,381	10,612,536	8,539,752	6.01
et	Estonian	226,552	548,589	419,373	2.42
eu	Basque	391,227	725,589	669,832	1.85
fa	Persian	892,984	2,012,489	1,748,880	2.25
fi	Finnish	528,323	2,938,331	1,916,372	5.56
fr	French	2,411,225	17,115,088	12,577,254	7.10
he	Hebrew	313,544	1,497,991	1,298,043	4.78
hr	Croatian	211,239	550,038	429,571	2.60
hu	Hungarian	501,758	2,241,596	1,646,175	4.47
hy	Armenian	291,266	1,853,522	1,294,452	6.36
id	Indonesian	618,676	2,170,068	1,700,961	3.51
it	Italian	1,748,062	7,769,065	5,780,364	4.44
ja	Japanese	1,319,693	12,153,736	8,237,546	9.21
kk	Kazakh	231,272	313,443	280,139	1.36
ko	Korean	584,594	1,599,714	1,327,504	2.74
lt	Lithuanian	202,444	486,654	447,025	2.40
ms	Malay	357,168	700,513	605,007	1.96
nl	Dutch	2,085,968	2,623,066	2,250,674	1.26
no	Norwegian (Bokmål)	582,399	1,874,697	1,490,498	3.22
pl	Polish	1,516,656	7,673,076	5,239,165	5.06
pt	Portuguese	1,088,286	6,636,422	5,116,972	6.10
ro	Romanian	428,682	2,021,351	1,327,598	4.72
ru	Russian	1,807,494	13,626,179	9,905,711	7.54
sh	Serbo-Croatian	456,444	1,368,842	909,406	3.00
simple	Simple English	207,354	630,729	515,962	3.04
sk	Slovak	240,027	562,559	456,986	2.34
sr	Serbian	657,077	3,234,971	1,760,098	4.92
sv	Swedish	2,580,001	11,695,159	7,875,678	4.53
tr	Turkish	477,885	2,216,325	1,567,293	4.64
uk	Ukrainian	1,146,175	4,291,799	3,457,589	3.74
vi	Vietnamese	1,271,057	3,392,140	2,846,216	2.67
zh	Chinese	1,264,023	6,730,567	5,182,993	5.32

separate language chapter). The highest value of this measure has English Wikipedia - almost 11 references per article. High values of RpA has also French (fr), Greek (el), Japanese (ja) and Russian (ru) Wikipedia.

IV. MODELS FOR WEB SOURCES

Based on previous study [15], this work used following models for sources assessment with changes (described in this section):

- 1) **F**-model – how frequently (*F*) of considered source appears in references.
- 2) **PR**-model – how popular (*P*) are Wikipedia articles in which considered source appears divided by number of the references (*R*) in such articles.
- 3) **AR**-model – how much authors (*A*) edited the articles in which considered source appears divided by number of the references (*R*) in such articles.

One of the most basic and commonly used approaches to assess the importance of a web source is to count how frequently it was used in Wikipedia articles. This principle

was used in relevant studies [26], [13], [27], [18]. So, **F**-model assesses how many times specific web domain occurs in external links of the references. For example, if the same source is cited 25 times in 13 Wikipedia articles (each contains at least one reference with such source), we count the (cumulative) frequency as 25. Equation 1 shows the calculation for **F**-model.

$$F(s) = \sum_{i=1}^n C_s(i), \quad (1)$$

where:

- s is the source, n is a number of the considered Wikipedia articles,
- $C_s(i)$ is a number of references using source s (e.q. domain in URL) in article i .

PR-model uses page views (or visits) of Wikipedia articles for certain period of time divided by the total number of all references in each considered Wikipedia article. Some studies showed correlation between information quality and page views in Wikipedia articles [28], [8], [29]. The more people read a specific Wikipedia article, the more likely its content was checked by part of them (including presence of reliable sources in references). So the more readers see the particular facts in the Wikipedia, the bigger probability that one of such reader will make appropriate edit if such facts are incorrect (or if source of information is inappropriate).

In other words, page views of the particular article usually shows the demand on information from Wikipedia readers. Therefore, visibility of the reference is also important. If more references are presented in the article, then the less visible is a specific source for the particular reader (visitor). At the same time, the more visitors has an Wikipedia article with references, the more visible is particular source in it. Equation 2 shows the calculation using **PR**-model.

$$PR(s) = \sum_{i=1}^n \frac{V(i)}{C(i)} \cdot C_s(i), \quad (2)$$

where:

- s is the source, n is a number of the considered Wikipedia articles,
- $C(i)$ is total number of the references in article i ,
- $C_s(i)$ is a number of the references using source s (e.q. domain in URL) in article i ,
- $V(i)$ is page views (visits) value of article for certain period of time i .

In comparison with previous research [15], for purposes of this study, apart from **PR**-model that uses cumulative page views V from humans (non-bots views) for a recent month (March 2022), additionally **PRy**-model will be used, which takes into account a wider date range - April 2021 - March 2022.

Quality of Wikipedia articles depends also on quantity and experience of authors who contributed to the content. Often articles in Wikipedia with the high quality are jointly created by a large number of different editors and this measure

positively correlates with information quality [30], [31], [32], [33], [29]. To assess popularity of an article from editing users there is a possibility to analyze revision history of the article to find how many authors were involved in content creation/editing. So, **AR**-model shows how popular article is among Wikipedia volunteer editors. Equation 3 presents this model in mathematical form.

$$AR(s) = \sum_{i=1}^n \frac{E(i)}{C(i)} \cdot C_s(i), \text{ where :} \quad (3)$$

- s is the source, n is a number of the considered Wikipedia articles,
- $C(i)$ is total number of the references in article i ,
- $C_s(i)$ is a number of the references using source s (e.q. domain in URL) in article i ,
- $E(i)$ is total number of authors of article i .

In contrast to previous work [15], **AR**-model in this study uses number of authors E that are registered on Wikipedia as users, without bot-users. Names of bots were selected based on the separate page (for example there is a special category in English Wikipedia [34]).

Additionally this study provides **ARe**-model, which is modification of **AR**-model: instead of counting the number of authors of a Wikipedia article, the number of editions of these authors (registered and non-bots) will be taken into account.

V. USING DBPEDIA AND WIKIDATA TO IDENTIFY WIKIPEDIA ARTICLES ABOUT COMPANIES

There are different possibilities to find topic of a particular Wikipedia article. For example, each article can be aligned to multiple categories, corresponding Wikidata item or DBpedia resource can highlight the topic based on properties in statements [29]. Additionally Wikipedia article can be included to different WikiProjects, that indicates interest to its information from groups of Wikipedia editors which focused on a specific topic (e.q. culture, history, military etc.).

This study used data from DBpedia and Wikidata to find Wikipedia articles related to companies. Each of those semantic databases have own advantages and disadvantages which are related to the operating principles and the technologies used.

A. DBpedia

DBpedia [35] is a semantic knowledge base that enriched automatically using structured information from Wikipedia articles in different languages [36], [37]. The resulting knowledge about some subject is available on the Web depending on title of Wikipedia article (as a source of that knowledge). For example, such semantic data about "Meta Platforms" as the DBpedia resource we can find on the page https://dbpedia.org/resource/Meta_Platforms because such data were extracted from the relevant article in English Wikipedia - https://en.wikipedia.org/wiki/Meta_Platforms. At the same time DBpedia has separate knowledge extracted from other language versions and we can find also relevant information on other pages extracted from other Wikipedia chapters. On

such DBpedia pages among different properties we can also find information about the type(s) of subject. In our example "Meta Platforms" aligned to "Company" and other classes of DBpedia ontology [38] and other structures. Such information is can generated automatically based on infoboxes (contained in Wikipedia articles) and their parameters. The figure 2 shows example of infoboxes about "Meta Platforms" company in different Wikipedia languages. DBpedia extracts information about infoboxes based on the source code (wiki code or wiki markup) of the Wikipedia articles.

DBpedia ontology has a hierarchical structure, and if some resource is aligned to other company-related classes, we can use connections between those classes to detect Wikipedia articles related to companies. For example, some of the organizations can be aligned to "Bank", "Publisher", "BusCompany" or other company-related class of DBpedia ontology, and after generalization we can find that all of them are belonging to "Company" class. Based on DBpedia dumps related to instance types [35] ("specific" part of the dumps for each available language) we found that Wikipedia articles can be aligned directly to one of 634 classes from DBpedia ontology. Figure 3 shows those classes distinguishing with larger font size the most popular ones: *Person*, *Species*, *PopulatedPlace*, *Insect*, *Settlement*, *Place* and other. "Company" class is the 20th most popular in such ranking.

It is worth mentioning that DBpedia provides two kinds of dumps that contain information on classification of resources (instances): instance-types (containing only direct types) and instance-types-transitive (containing the transitive types of a resource based on the DBpedia ontology). Such files contain triples of the form '*<resource> rdf:type <class>*' generated by the mappings extraction and other techniques for different language chapters of Wikipedia.

Figure 4 shows the structure of a part of DBpedia ontology with "Organisation" class as a root node. It also presents information about directly alignments to separate classes of this ontology based on English Wikipedia. We can find there numbers based on of instances-types (direct alignment).

If we include also information on transitive types, we will have more resources aligned to same classes by taking into account connections between them in the DBpedia ontology. Figure 5 shows those classes distinguishing with larger font size the most popular ones: *Species*, *Eukaryote*, *Animal*, *Person*, *Location*, *Place* and other. "Company" class is the 34th most popular in such ranking.

After considering transitive DBpedia dumps we have got additionally 20,736 resources (to directly aligned 64,372 resources) in "Company" class - 85,108 in total in that class based on data from English Wikipedia. Next we took similar data extracted by DBpedia from other Wikipedia languages, and finally we got 173,418 unique companies ¹. Further we

¹Unique company in this case means, that separate Wikipedia articles in various languages related to the same company counted as 1 company (instead of counting each Wikipedia article in each language version as a separate company).

used "DBO-companies" for the obtained list of Wikipedia articles about companies based on DBpedia extraction.

B. Wikidata

Wikidata [40] is a semantic knowledge base that works on a similar principles that Wikipedia with one important difference - here we can insert facts about the subjects using statements with properties and values rather than sentences in natural language. Wikidata is also considered as the central data management platform for Wikipedia and most of its sister projects [41].

Each Wikidata item has a collection of different statements structured in the form: "Subject-Predicate-Object". Figure 6 shows Wikidata item Q380 ("Meta Platforms") with some statements.

Based on Wikidata statements we can find items on a specific topic. In our case, we will use the statement "Property:P31 Q783794" ("instance of" - "company"). Listing 1 presents SPARQL query to get such list from Wikidata using its query service [43]. Result of this query is available on the web page: <https://w.wiki/5Bsc>.

```
SELECT ?item WHERE {
    ?item wdt:P31 wd:Q783794 . }
```

Listing 1. SPARQL query to get list of Wikidata items directly connected to "company" item (Q783794) by "instance of" property (P31)

So, based on simple query we have got 12,635 Wikidata items related to companies. However, there are other connections in Wikidata that indicate items related to our topic. Similarly to DBpedia, here we can have also other "sub-classes" or alternatives that can build more complete list of Wikidata items which can give list of appropriate Wikipedia articles. Let's go back to our example on "Meta Platforms" as an Wikidata item showed on the figure 6. We can see, that apart from "company", this item is also aligned to "business" (Q4830453), "enterprise" (Q6881511), "public company" (Q891723) and "technology company" (Q18388277) by "instance of" parameter. Now we will use this information to enrich our query - listing 2 presents such SPARQL query: <https://w.wiki/5Bsw>. This query returned much more Wikidata items (comparing previous one) - 275,944 items. It is important to note, that this number doesn't show directly number of Wikipedia articles related to companies, because not all Wikidata items contains links to at least one Wikipedia article.

```
SELECT ?item WHERE {
VALUES ?com {wd:Q783794 wd:Q4830453
    wd:Q6881511 wd:Q891723 wd:Q18388277}
?item wdt:P31 ?com . }
```

Listing 2. SPARQL query to get list of Wikidata items directly connected to "company" (Q783794), business (Q4830453), enterprise (Q6881511), "public company" (Q891723) and "technology company" (Q18388277) by "instance of" property (P31)

Despite significant increase of Wikidata items based on more complex query, there can be at least one important questions: is the proposed query complete enough to find all (or at least most of) Wikidata items related to companies?

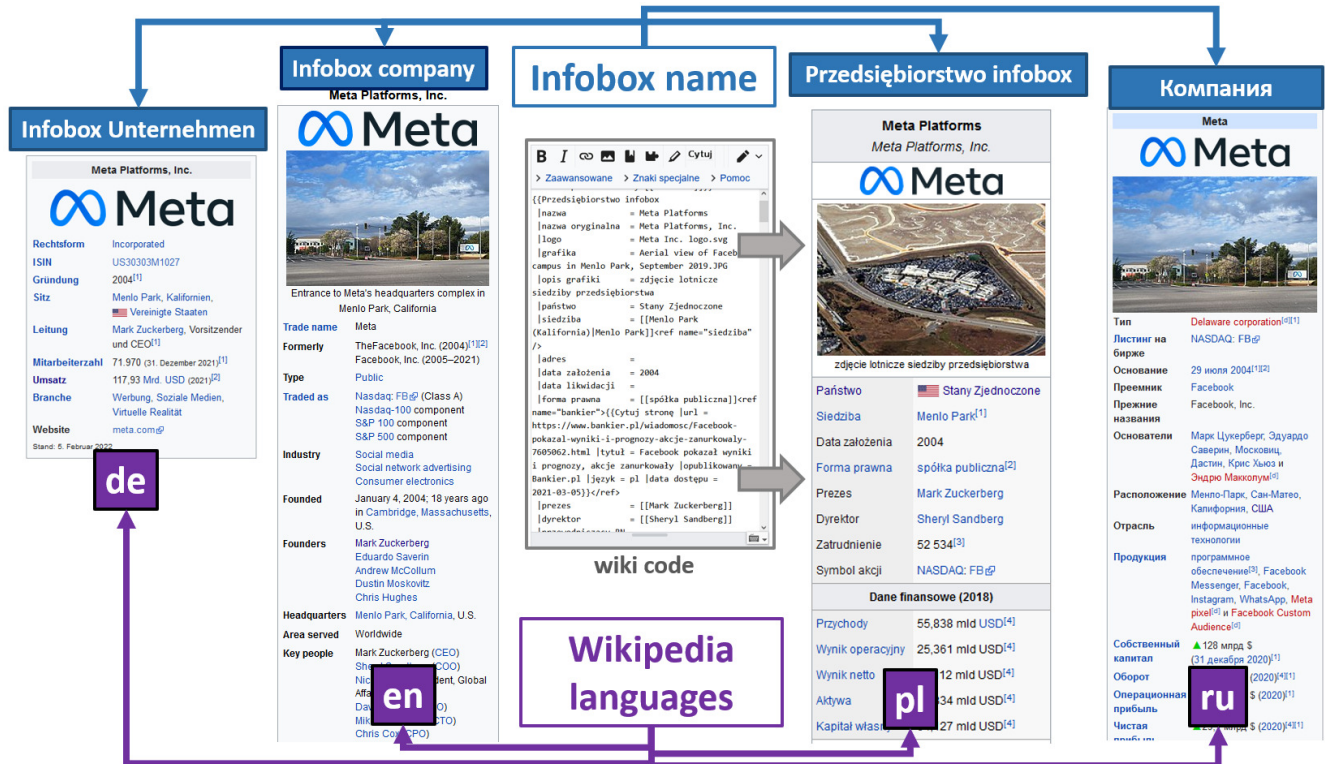


Fig. 2. Infoboxes about "Meta Platforms" company in different Wikipedia chapters



Fig. 3. Popular DBpedia ontology classes that are directly aligned to resources in various languages. Source: own calculations based on DBpedia ontology instance types specific dumps [35].

First, let's try to obtain general statistics on values that are inserted to "instance of" (P31) parameter among over 95 million Wikidata items. To do so, we prepared special algorithm in Python to extract such information from Wikidata

dumps in JSON format [44]. It is worth noticing, that it is possible to construct SPARQL query to solve this task, however due to limitation of the Wikidata query service (such as limited time execution of the query) such statistics and other complex analysis can be done by extracting necessary data from the dump files. Figure 7 shows those items distinguishing with larger font size the most popular ones: *scholarly article* (Q13442814), *human* (Q5), *Wikimedia category* (Q4167836), *temporal range start* (Q523), *Taxon* (Q16521), *infrared source* (Q67206691), *galaxy* (Q318) and other. Overall there are 87501 different alignments ("classes"). Items related to companies, such as "business" (Q4830453), "enterprise" (Q6881511) are on the 39th, 129th place respectively in such ranking.

Next we conduct such analysis only on Wikidata items, which has at least one link to Wikipedia article of one of the 42 considered languages in this study (see table I). Results are shown in figure 8. Now we have got 67,634 different alignments ("classes") and on the top we have: *Wikimedia category* (Q4167836), *human* (Q5), *Wikimedia disambiguation page* (Q4167410), *Wikimedia template* (Q11266439), *human settlement* (Q486972), *Wikimedia list article* (Q13406463), *album* (Q482994), *film* (Q11424), *village* (Q532) and others. Early considered items related to companies now are higher in the ranking: "business" (Q4830453) took 12th place, "enterprise" (Q6881511) took 66th place.

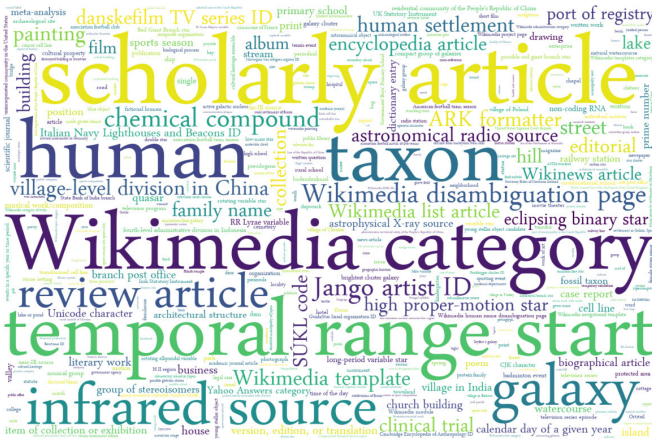


Fig. 7. Popular Wikidata items as a values in "instance of" statements. Source: own calculations based on Wikidata dumps files [44].

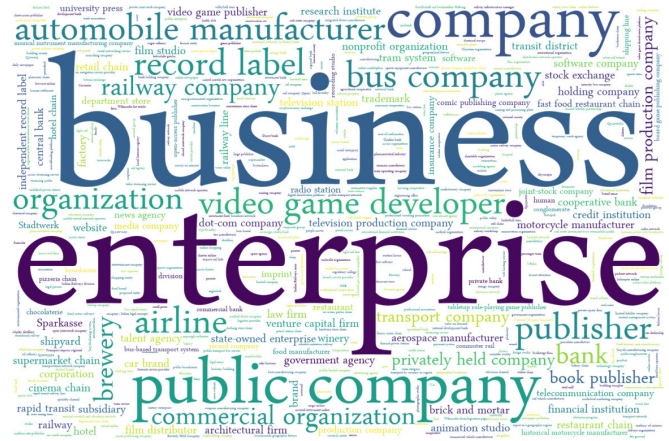


Fig. 9. Popular Wikidata items as a values in "instance of" statements. Only Wikidata items with link to at least one Wikipedia article related to DBO-companies. Source: own calculations based on Wikidata dumps files [44].



Fig. 8. Popular Wikidata items as a values in "instance of" statements. Only Wikidata items with at least one link to Wikipedia article from one of 42 considered languages. Source: own calculations based on Wikidata dumps files [44].

at least 200 times to avoid insignificant mistakes that could be done by some users that edit Wikidata. In that case we will have 63 Wikidata items, that can appear in "instance of" (P31) statements as a values. Additionally we removed alignment to "organization" (Q43229) which is too general.

As a result, we have more Wikidata items with articles on the list of companies - overall 291,768 Wikidata items with at least one related Wikipedia article in considered language versions were identified. In further analysis we will use "WCA-companies" for this list.

VI. ESTIMATING THE INFORMATION SOURCES IN WIKIPEDIA ABOUT COMPANIES

This section presents results of assessment of the most important sources of information companies across Wikipedia languages using different models.

Due to the limitation of space, following subsections presents results for the 15 most developed language versions of Wikipedia (with at least 1 million articles, see table I) Additionally, for the charts below, only the websites that appear at least 20 times in the top 100 at each language/model intersection² were selected. The more extended and interactive results can be found in supplementary materials [39].

It is important to note that archive services (such as archive.org) were excluded from analysis, due to the frequent occurrence of such links alongside the original sources in the same reference. If original source is no longer available, such archive services are very important, because Wikipedia readers can verify information, but unavailable original web sources are not a scope of this research. References to Wikipedia itself and Wikidata were also excluded. Links that are automatically inserted to references based on such identifiers as DOI (often links to doi.org) or ISBN (often links to books.google.com) cannot indicate directly the source of information. So such links were not considered in website analysis.

A. DBO-companies

First, we conducted a source analysis for the list of Wikipedia articles that have been generated based on data from DBpedia (see V-A) - "DBO-companies". Figure 10 shows the most important web sources of information on companies described in Wikipedia based with positions in rankings across 15 most developed language versions using five considered models.

Top 10 web sources in DBO-companies across 15 considered languages according to different models are as follows: nytimes.com, reuters.com, youtube.com, bloomberg.com, forbes.com, techcrunch.com, bbc.co.uk, cnn.com, wsj.com, theguardian.com.

²15 languages and 5 models gives 75 such intersections

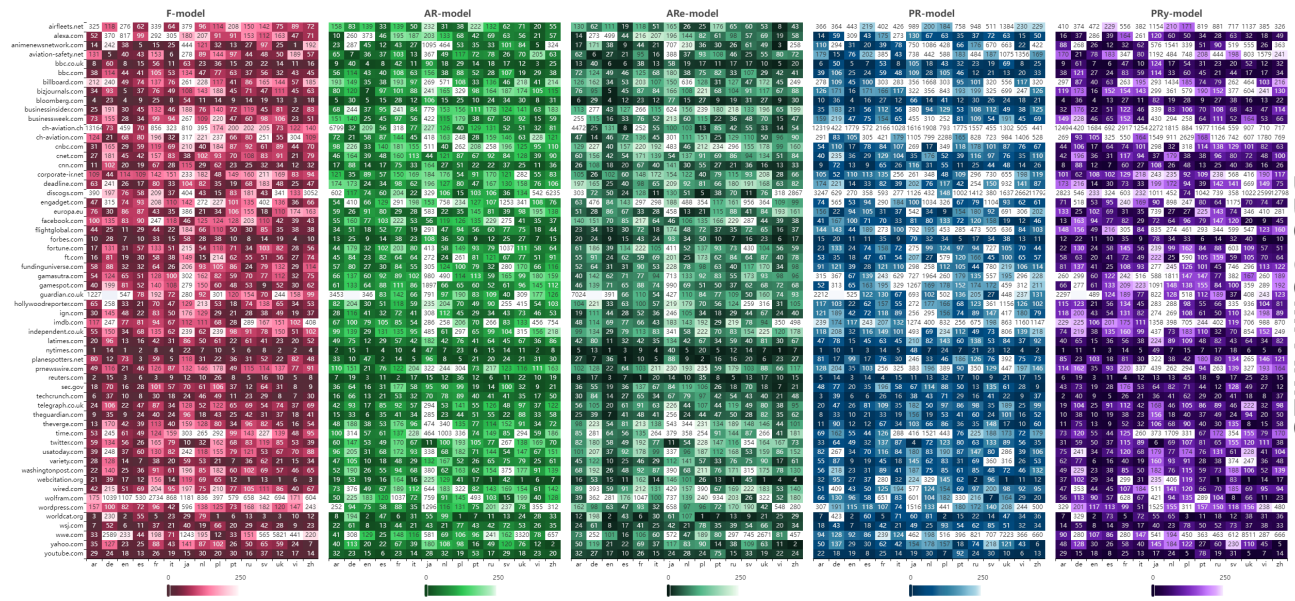


Fig. 10. The most important web sources of information on companies described in Wikipedia based on "DBO-companies" list with positions in rankings across 15 most developed language versions using various models. Source: own calculation based on Wikimedia dumps in April 2022. More extended and interactive version of the heat maps is available in [39]

B. WCA-companies

Figure 11 presents the most important web sources of information on companies described in Wikipedia based on WCA-companies list (described in V-B and V-C) with positions in rankings across 15 most developed language versions using five considered models.

Top 10 web sources in WCA-companies across 15 considered languages according to different models are as follows: nytimes.com, reuters.com, youtube.com, techcrunch.com, forbes.com, bloomberg.com, bbc.co.uk, theguardian.com, wsj.com, cnn.com.

C. Wikipedia languages

Based on average position in rankings calculated using different models we prepared the top 10 most important sources of information about companies in each Wikipedia languages.

Lists of such sources are presented below.

- **Arabic Wikipedia (ar):** grid.ac, nytimes.com, worldcat.org, alexa.com, bbc.co.uk, bloomberg.com, reuters.com, techcrunch.com, theguardian.com, cnn.com
- **Belarusian Wikipedia (be):** webcitation.org, tut.by, belta.by, zvi-azda.by, svaboda.org, nrbf.by, alexa.com, sec.gov, europa.eu, worldcat.org
- **Bulgarian Wikipedia (bg):** capital.bg, brra.bg, dnevnik.bg, webcitation.org, alexa.com, bbc.co.uk, nytimes.com, forbes.com, vesti.bg, q4cdn.com
- **Catalan Wikipedia (ca):** gencat.cat, elpais.com, worldcat.org, enciclopedia.cat, lavanguardia.com, ara.cat, vilaweb.cat, nytimes.com, elpuntavui.cat, elmundo.es
- **Czech Wikipedia (cs):** idnes.cz, justice.cz, worldcat.org, ihned.cz, lupa.cz, novinky.cz, denik.cz, ceskatelevize.cz, e15.cz, zdopravy.cz
- **Danish Wikipedia (da):** dr.dk, business.dk, politiken.dk, borsen.dk, finans.dk, computerworld.dk, berlingske.dk, tv2.dk, ing.dk, nytimes.com

- **German Wikipedia (de):** spiegel.de, zdb-katalog.de, handelsblatt.com, mementoweb.org, heise.de, welt.de, faz.net, sueddeutsche.de, zeit.de, nytimes.com
- **Greek Wikipedia (el):** et.gr, kathimerini.gr, tovima.gr, reuters.com, bbc.co.uk, capital.gr, nytimes.com, youtube.com, worldcat.org, typologies.gr
- **English Wikipedia (en):** nytimes.com, worldcat.org, reuters.com, bbc.co.uk, bloomberg.com, theguardian.com, wsj.com, bizjournals.com, forbes.com, indiatimes.com
- **Esperanto Wikipedia (eo):** staralliance.com, webcitation.org, liberfolio.org, wikimedia.org, wikiwix.com, nytimes.com, vortaro.net, debian.org, elpais.com, bloomberg.com
- **Spanish Wikipedia (es):** elpais.com, issn.org, nytimes.com, elmundo.es, youtube.com, bbc.co.uk, lanacion.com.ar, planespotters.net, reuters.com, abc.es
- **Estonian Wikipedia (et):** postimees.ee, err.ee, delfi.ee, riigiteataja.ee, aripaev.ee, muinas.ee, digar.ee, dv.ee, nasdaqbaltic.com, inforegister.ee
- **Basque Wikipedia (eu):** berria.eus, worldcat.org, argia.eus, elpais.com, euskadi.net, euskadi.eus, eitb.eus, nih.gov, berria.info, diariavasco.com
- **Persian Wikipedia (fa):** bbc.co.uk, bbc.com, webcitation.org, reuters.com, nytimes.com, sec.gov, forbes.com, alexa.com, isna.ir, radiofarda.com
- **Finnish Wikipedia (fi):** yle.fi, hs.fi, kauppaletti.fi, is.fi, forbes.com, talouselama.fi, bloomberg.com, iltalehti.fi, taloussanomati.fi, nytimes.com
- **French Wikipedia (fr):** lesechos.fr, lemonde.fr, reuters.com, lefigaro.fr, worldcat.org, societe.com, zonebourse.com, wikiwix.com, liberation.fr, lexxpress.fr
- **Hebrew Wikipedia (he):** globes.co.il, themarker.com, nli.org.il, ynet.co.il, calcalist.co.il, haaretz.co.il, walla.co.il, tase.co.il, mako.co.il, nytimes.com
- **Croatian Wikipedia (hr):** bbc.co.uk, vecernji.hr, hrt.hr, zse.hr, tportal.hr, nytimes.com, enciklopedija.hr, jutarnji.hr, poslovi.hr, alexa.com
- **Hungarian Wikipedia (hu):** index.hu, origo.hu, hvg.hu, youtube.com, nytimes.com, blog.hu, iho.hu, crt-tv.com, 24.hu, napi.hu
- **Armenian Wikipedia (hy):** webcitation.org, nytimes.com, youtube.com, bbc.co.uk, sec.gov, purl.org, wsj.com, vedomosti.ru, kommersant.ru, forbes.com
- **Indonesian Wikipedia (id):** detik.com, Kompas.com, nytimes.com, forbes.com, worldcat.org, tempo.co, alexa.com, bbc.co.uk, reuters.com, kontan.co.id

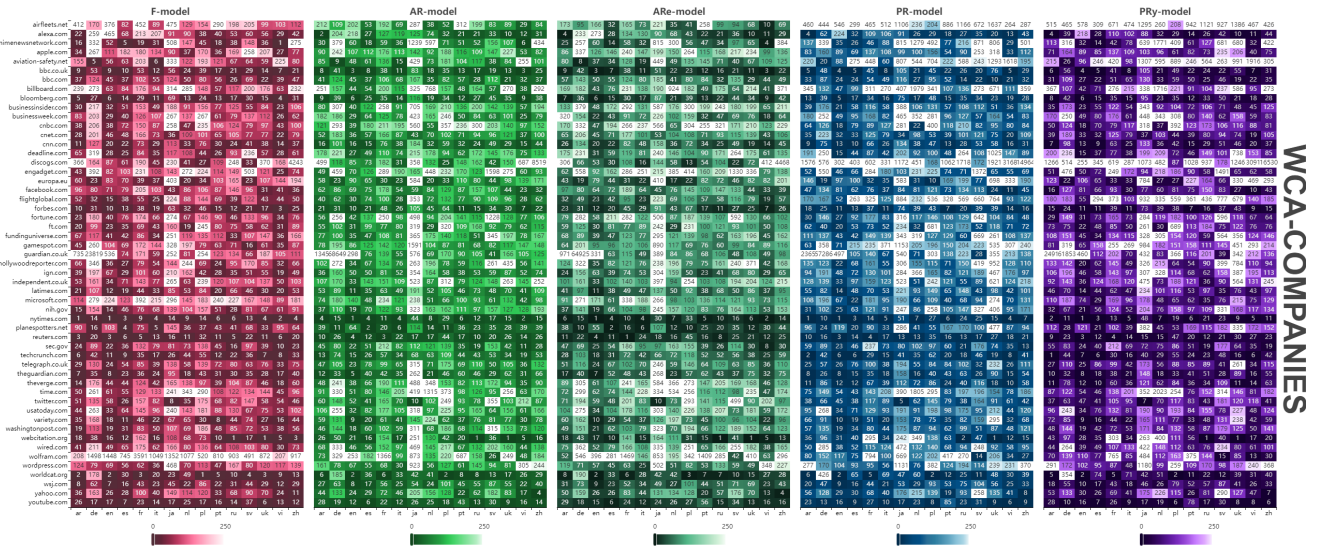


Fig. 11. The most important web sources of information on companies described in Wikipedia based on "WCA-companies" list with positions in rankings across 15 most developed language versions using various models. Source: own calculation based on Wikimedia dumps in April 2022. More extended and interactive version of the heat map is available in [39]

- **Italian Wikipedia (it):** repubblica.it, corriere.it, ilsolo24ore.com, nytimes.com, ansa.it, lastampa.it, bbc.co.uk, youtube.com, treccani.it, primaonline.it
- **Japanese Wikipedia (ja):** catr.jp, nikkei.com, ndl.go.jp, impress.co.jp, asahi.com, itmedia.co.jp, twitter.com, eir-parts.net, edinet-fsa.go.jp, prtmes.jp
- **Kazakh Wikipedia (kk):** webcitation.org, sec.gov, kase.kz, tengrinews.kz, bbc.co.uk, nytimes.com, lenta.ru, vedomosti.ru, shareholder.com, railways.kz
- **Korean Wikipedia (ko):** naver.com, chosun.com, mt.co.kr, hankyung.com, mk.co.kr, donga.com, yonhapnews.co.kr, hani.co.kr, asiae.co.kr, khan.co.kr
- **Lithuanian Wikipedia (lt):** vz.lt, delfi.lt, 15min.lt, vie.lt, bloomberg.com, lrytas.lt, ft.com, lrs.lt, lrt.lt, bbc.co.uk
- **Malay Wikipedia (ms):** thestar.com.my, nytimes.com, bloomberg.com, sec.gov, utusan.com.my, forbes.com, reuters.com, worldcat.org, cnn.com, bbc.co.uk
- **Dutch Wikipedia (nl):** nrc.nl, volkskrant.nl, nu.nl, nos.nl, fd.nl, standaard.be, telegraaf.nl, nytimes.com, ad.nl, kb.nl
- **Norwegian (Bokmål) Wikipedia (no):** nb.no, nrk.no, breg.no, e24.no, regjeringen.no, aftenposten.no, dn.no, snl.no, proff.no, vg.no
- **Polish Wikipedia (pl):** wirtualnedia.pl, worldcat.org, wyborcza.pl, sejm.gov.pl, satkuriel.pl, pwn.pl, rynek-kolejowy.pl, rp.pl, onet.pl, wp.pl
- **Portuguese Wikipedia (pt):** uol.com.br, globo.com, abril.com.br, estadao.com.br, nytimes.com, worldcat.org, sapo.pt, forbes.com, terra.com.br, bloomberg.com
- **Romanian Wikipedia (ro):** zf.ro, wall-street.ro, money.ro, adevarul.ro, capital.ro, mediafax.ro, evz.ro, hotnews.ro, nytimes.com, romanialibera.ro
- **Russian Wikipedia (ru):** webcitation.org, kommersant.ru, vedomosti.ru, rbc.ru, lenta.ru, ria.ru, forbes.ru, tass.ru, reuters.com, cnews.ru
- **Serbo-Croatian Wikipedia (sh):** nytimes.com, cnn.com, worldcat.org, bbc.co.uk, britannica.com, rts.rs, yahoo.com, washingtonpost.com, alexa.com, nih.gov
- **Simple English Wikipedia (simple):** nytimes.com, wolfram.com, mathvault.ca, worldcat.org, bbc.co.uk, latimes.com, bloomberg.com, yahoo.com, reuters.com, sec.gov
- **Slovak Wikipedia (sk):** worldcat.org, sme.sk, dennik.sk, finstat.sk, etrend.sk, hnonline.sk, orsr.sk, aktualy.sk, pravda.sk, idnes.cz
- **Serbian Wikipedia (sr):** b92.net, rts.rs, alexa.com, worldcat.org, nytimes.com, novosti.rs, politika.rs, apr.gov.rs, bbc.co.uk, blic.rs
- **Swedish Wikipedia (sv):** allabolag.se, svd.se, dn.se, kb.se, svt.se, di.se, idg.se, mynewsdesk.com, worldcat.org, ne.se
- **Turkish Wikipedia (tr):** hurriyet.com.tr, milliyet.com.tr, nytimes.com, haberturk.com, techcrunch.com, alexa.com, sec.gov, sabah.com.tr, youtube.com, bloomberg.com
- **Ukrainian Wikipedia (uk):** webcitation.org, rada.gov.ua, rbc.ua, epravda.com.ua, pravda.com.ua, uprom.info, youtube.com, ukrain-form.ua, nytimes.com, detector.media
- **Vietnamese Wikipedia (vi):** nytimes.com, vnexpress.net, bbc.co.uk, tuoitre.vn, forbes.com, webcitation.org, bloomberg.com, youtube.com, techcrunch.com, animenewsnetwork.com
- **Chinese Wikipedia (zh):** sina.com.cn, xinhuanet.com, qq.com, ltn.com.tw, yahoo.com, udn.com, sohu.com, chinatimes.com, nytimes.com, youtube.com

VII. CONCLUSION AND FUTURE WORK

This study focused on information sources analysis of Wikipedia about companies in different languages. After extraction over 230 million references there were a process of identification of the main websites address for each URL address. As a result - over 2 million unique websites have been identified. To find important web sources across the languages, topics of the Wikipedia articles were analyzed. Using semantic representation of those information in DBpedia and user-generated knowledge in Wikidata this study shows how to find important web sources across languages based on existing and new models.

Models presented in this work can help not only Wikipedia volunteer editors to select web sites that can provide valuable information on companies, but also can help other Internet users to better understand how to find valuable sources of information a specific topic on the Web using open data from Wikipedia.

We plan to extend this research in future by providing additional features on identification of companies in Wikipedia. Additionally, we plan to provide different organizations to specific sectors (industries) to find the differences between reliability of information sources.

Future work will be focused also on extending reliability models and using different methods on topic classifications. One of the directions is to develop ways of weighting the importance of a reference based on its position within a Wikipedia article. There are also plans on including different measures related to the reputation of Wikipedia authors, protection of the articles, topic similarity and others.

ACKNOWLEDGEMENTS

The project financed within the Regional Initiative for Excellence programme of the Minister of Education and Science of Poland, years 2019-2023, grant no. 004/RID/2018/19, financing 3,000,000 PLN.

REFERENCES

- [1] English Wikipedia, "Wikipedia:Reliable sources," https://en.wikipedia.org/wiki/Wikipedia:Reliable_sources, 2022.
- [2] —, "Wikipedia:Verifiability," <https://en.wikipedia.org/wiki/Wikipedia:Verifiability>, 2022.
- [3] —, "Wikipedia:Reliable sources/Perennial sources," https://en.wikipedia.org/wiki/Wikipedia:Reliable_sources/Perennial_sources, 2022.
- [4] Internet Live Stats, "Total number of Websites," <https://www.internetlivestats.com/total-number-of-websites/>, 2022.
- [5] Netcraft, "August 2021 Web Server Survey," <https://news.netcraft.com/archives/2021/08/25/august-2021-web-server-survey.html>, 2021.
- [6] B. Stvilia, M. B. Twidale, L. C. Smith, and L. Gasser, "Assessing information quality of a community-based encyclopedia," *Proc. ICIQ*, pp. 442–454, 2005.
- [7] J. E. Blumenstock, "Size matters: word count as a measure of quality on Wikipedia," in *Proceedings of the 17th international conference on World Wide Web*. ACM, 2008, pp. 1095–1096.
- [8] W. Lewoniewski, "The method of comparing and enriching information in multilingual wikis based on the analysis of their quality," PhD, Poznań University of Economics and Business, 2018. [Online]. Available: http://www.wbc.poznan.pl/Content/461699/Lewoniewski_Wlodzimierz-rozprawa_doktorska.pdf
- [9] WikiRank, "Quality and Popularity Assessment of Wikipedia Articles," <https://wikirank.net/>, 2022.
- [10] P. Tzekou, S. Stamou, N. Kirtsis, and N. Zotos, "Quality Assessment of Wikipedia External Links," in *WEBIST*, 2011, pp. 248–254.
- [11] E. Yaari, S. Baruchson-Arbib, and J. Bar-Ilan, "Information quality assessment of community generated content: A user study of Wikipedia," *Journal of Information Science*, vol. 37, no. 5, pp. 487–498, 2011.
- [12] R. Conti, E. Marzini, A. Spognardi, I. Matteucci, P. Mori, and M. Petrocchi, "Maturity assessment of Wikipedia medical articles," in *Computer-Based Medical Systems (CBMS), 2014 IEEE 27th International Symposium on*. IEEE, 2014, pp. 281–286.
- [13] W. Lewoniewski, K. Węcel, and W. Abramowicz, "Analysis of references across Wikipedia languages," in *International Conference on Information and Software Technologies*. Springer, 2017, pp. 561–573.
- [14] F. Å. Nielsen, D. Mietchen, and E. Willighagen, "Scholia, scientometrics and Wikidata," in *European Semantic Web Conference*. Springer, 2017, pp. 237–259.
- [15] W. Lewoniewski, K. Węcel, and W. Abramowicz, "Modeling Popularity and Reliability of Sources in Multilingual Wikipedia," *Information*, vol. 11, no. 5, p. 263, 2020.
- [16] H. Singh, R. West, and G. Colavizza, "Wikipedia citations: A comprehensive data set of citations with identifiers extracted from English Wikipedia," *Quantitative Science Studies*, vol. 2, no. 1, pp. 1–19, 2021.
- [17] M. Teplitskiy, G. Lu, and E. Duede, "Amplifying the impact of open access: Wikipedia and the diffusion of science," *Journal of the Association for Information Science and Technology*, vol. 68, no. 9, pp. 2116–2127, 2017.
- [18] D. Jemielniak, G. Masukume, and M. Wilamowski, "The most influential medical journals according to Wikipedia: quantitative analysis," *Journal of medical Internet research*, vol. 21, no. 1, p. e11429, 2019.
- [19] G. Colavizza, "COVID-19 research in Wikipedia," *Quantitative Science Studies*, vol. 1, no. 4, pp. 1349–1380, 12 2020. [Online]. Available: https://doi.org/10.1162/qss_a_00080
- [20] B. Fetahu, K. Markert, W. Nejdil, and A. Anand, "Finding news citations for wikipedia," in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, 2016, pp. 337–346.
- [21] T. Piccardi, M. Redi, G. Colavizza, and R. West, "Quantifying engagement with citations on Wikipedia," in *Proceedings of The Web Conference 2020*, 2020, pp. 2365–2376.
- [22] BestRef, "Popularity and Reliability Assessment of Wikipedia Sources," <https://bestref.net>, 2022.
- [23] Wikimedia Downloads, "Main page," <https://dumps.wikimedia.org>, 2021.
- [24] W. Lewoniewski, K. Węcel, and W. Abramowicz, "Reliability in Time: Evaluating the Web Sources of Information on COVID-19 in Wikipedia across Various Language Editions from the Beginning of the Pandemic," 2022, presented at Wiki WorkShop 2022 (held virtually at The Web Conference 2022) on April 25, 2022.
- [25] Public Suffix List, "List," <https://publicsuffix.org/learn/>, 2021.
- [26] F. Å. Nielsen, "Scientific citations in Wikipedia," *arXiv preprint arXiv:0705.2106*, 2007.
- [27] M. Redi, "Characterizing Wikipedia Citation Usage. Analyzing Reading Sessions," https://meta.wikimedia.org/wiki/Research:Characterizing_Wikipedia_Citation_Usage/Analyzing_Reading_Sessions, 2019, [Online; accessed 01-Sep-2021].
- [28] J. Lerner and A. Lomi, "Knowledge categorization affects popularity and quality of Wikipedia articles," *PLoS one*, vol. 13, no. 1, p. e0190674, 2018.
- [29] W. Lewoniewski, K. Węcel, and W. Abramowicz, "Multilingual Ranking of Wikipedia Articles with Quality and Popularity Assessment in Different Topics," *Computers*, vol. 8, no. 3, 2019. [Online]. Available: <https://www.mdpi.com/2073-431X/8/3/60>
- [30] A. Lih, "Wikipedia as Participatory Journalism: Reliable Sources? Metrics for evaluating collaborative media as a news resource," *5th International Symposium on Online Journalism*, p. 31, 2004.
- [31] D. M. Wilkinson and B. a. Huberman, "Cooperation and quality in wikipedia," *Proceedings of the 2007 international symposium on Wikis WikiSym 07*, pp. 157–164, 2007.
- [32] G. C. Kane, "A multimethod study of information quality in wiki collaboration," *ACM Transactions on Management Information Systems (TMIS)*, vol. 2, no. 1, p. 4, 2011.
- [33] J. Liu and S. Ram, "Using big data and network analysis to understand Wikipedia article quality," *Data & Knowledge Engineering*, 2018.
- [34] English Wikipedia, "Category:All Wikipedia bots," https://en.wikipedia.org/wiki/Category:All_Wikipedia_bots, 2022.
- [35] Databus, "DBpedia Ontology instance types," <https://databus.dbpedia.org/dbpedia/mappings/instance-types/>, 2022.
- [36] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, "Dbpedia: A nucleus for a web of open data," in *The semantic web*. Springer, 2007, pp. 722–735.
- [37] J. Lehmann, R. Isele, M. Jakob, A. Jentzsch, D. Kontokostas, P. N. Mendes, S. Hellmann, M. Morsey, P. Van Kleef, S. Auer *et al.*, "Dbpedia—a large-scale, multilingual knowledge base extracted from wikipedia," *Semantic web*, vol. 6, no. 2, pp. 167–195, 2015.
- [38] DBpedia, "Ontology Classes," <http://mappings.dbpedia.org/server/ontology/classes/>, 2022.
- [39] data.lewoniewski.info, "Supplementary materials for this research," <https://data.lewoniewski.info/companies/>, 2022.
- [40] Wikidata, "Main Page," https://www.wikidata.org/wiki/Wikidata:Main_Page, 2022.
- [41] D. Vrandečić and M. Krötzsch, "Wikidata: a free collaborative knowledgebase," *Communications of the ACM*, vol. 57, no. 10, pp. 78–85, 2014.
- [42] Wikidata, "Q380," <https://www.wikidata.org/wiki/Q380>, 2022.
- [43] Wikidata Query Service, "Main page," <https://query.wikidata.org/>, 2022.
- [44] Wikimedia Downloads, "Wikidata Wiki Entities," <https://dumps.wikimedia.org/wikidatawiki/entities/>, 2022.

Modelling an IT solution to anonymise selected data processed in digital documents

Barbara Probiez^{*}, Tomasz Jach^{*}, Jan Kozak^{*}, Radosław Pacud[†] and Tomasz Turek[‡]

^{*} Department of Machine Learning,
 University of Economics in Katowice,
 1 Maja, 40-287 Katowice, Poland

Email: barbara.probiez,tomasz.jach,jan.kozak@ue.katowice.pl

[†] Faculty of Finance,
 University of Economics in Katowice,
 1 Maja, 40-287 Katowice, Poland

Email: radoslaw.pacud@ue.katowice.pl

[‡] Faculty of Management,
 Częstochowa University of Technology,
 al. Armii Krajowej 19 B, 42-201 Częstochowa, Poland
 Email: tomasz.turek@pcz.pl

Abstract—Allowing access to real legal documents is an important element both for the development of science and the judiciary. On the other hand, protecting information about citizens or organizations, that appear in these documents, is crucial and required by law. Therefore, before the documents are distributed, the data anonymisation process should be carried out. Unfortunately, there is no perfect tool that can automatically anonymise documents in such a way, that the main concept of the document is preserved; especially in the case of documents written in inflectional language. The aim of this article is to show how important (and at the same time how difficult) is the task to identify personal or corporate data of a client, as well as other related personal data in documents that are subject to legal protection. We conducted research aimed at assessing the usefulness of IT techniques as well as decision rules and patterns in the anonymisation of legal documents. A set of real legal documents written in Polish was used for the research in which we identified selected types of data that need to be anonymised. Eventually, the obtained results were assessed by field experts. Additionally, in order to verify the effectiveness of the proposed solution, we conducted research on a set of 50,000 false identities with names, company names, addresses and other confidential information. The collection was created using Fake Name Generator¹. The obtained results from both experiments confirmed that the solutions we proposed is accurate even in the case of real legal documents.

I. INTRODUCTION AND RELATED WORKS

THE PROCESS of anonymising documents is important to protect individuals and institutions from the illegal dissemination of information [1]. However, in the case of legal documents, this is a very difficult process due to unstructured content of the documents [2]. In addition, the provisions of law, i.e. the GDPR [3], will force institutions to appropriately transform their current IT systems in the field of personal data processing. Due to the large variety of document structures and the lack of a uniform system applicable to all judicial

organizations, there is a high risk of privacy breach. Due to the variety of document structures and the lack of a uniform system applicable to all judicial organizations, there is a high risk of privacy breach. For this reason, the anonymisation process is often widely regarded as an expensive and inefficient process [4]. Additionally, due to the low effectiveness of the available data anonymisation tools, the anonymisation process in law firms is most often performed manually by trained employees [5].

In this article, we will look at the theoretical assumptions of a basic IT tool designed to identify selected types of data in text documents. On the other hand, the research goal is to check and evaluate the usefulness of IT techniques as well as decision rules and patterns in the anonymisation of legal documents. In the case of legal documents, data identifying individuals or organizations must be removed, anonymised or pseudonymised from the digital document following the end of the legal service. Authors are convinced that if such data are not manually obliterated - either manually or using a computer-assisted method - any lawyer should delete all documents of clients from law firm's digital media and computers in a time demanded by national regulation of legal occupation or implicit period of time assumed by the professional need (processing by the purpose). By all above mentioned regulations personal data and other types must be deleted or pseudonymised by legal tech. To achieve this goal, a team of coauthors is working on a project designed to build such a technology².

The first step in anonymising text documents is the ex-

²Project developed by Infojura Sp. z o. o. (KRS 0000502117) carried out in the period from 01/01/2022 to 31/12/2023 under the name "Technologies for Automatic anonymisation of Personal Data" in digital documents. It is financed with the support of EU as part of the Regional Operational Program of the Silesian Voivodeship for 2014-2020 (European Regional Development Fund). Action 1.2. Research, development and innovation in enterprises.

¹<https://www.fakenamegenerator.com/>

traction of information that directly identifies individuals or organizations. They are referred to as Named Entities (NE) [6]. Most often, NE are classified into predefined semantic categories, i.e. first name, last name, organization name or location [7], [8]. Natural language processing (NLP) methods are used to automatically recognize NE [9]. For this purpose, Named Entity Recognition (NER) [10] was created. It is the process of locating information in the text, which then becomes a specific category. In addition to the basic categories, i.e. the name of an organization or person, NER has added categories that define time and numerical expressions, such as monetary values [11].

In the process of identifying named entities, the language in which the text document is written is very important. In the case of languages with an extensive morphological structure, the processing of text documents and the extraction of information is very difficult [12]. Therefore, Graliński et al [13] presented a formalism for the rule-based NER. Their research focuses on the use of NER for inflectional languages (especially for Polish and Czech) and the translation of Named Entities. Researchers developed two applications that could be used for machine anonymisation and machine translation. Similarly, J. Pisowski [14] created the formalism of recognizing NE from Polish texts and manually created a set of NER rules for the Polish language.

To assist researchers in sharing raw textual data, Kleinberg et al. [15] proposed the NETANOS anonymisation system that identifies and modifies named entities (e.g. people, locations, times, dates). Bayesian tests showed that NETANOS anonymisation was practically equivalent to human anonymisation. The authors only used NER to detect personal data. Using this method, they hypothesized that the ability to discover the original meaning of anonymised data could ultimately be similar to that of humans if the training data set is large enough.

Unfortunately, the NER in the field of law, despite its importance, is not a well-researched area. Many current approaches use different techniques and classification methods on different datasets [16]. Additionally, most of the obtained anonymisation results are only assessed by experts. For this reason, it is not possible to properly compare the results. However, the proposed solutions make a significant contribution and are the basis for further research. It should be remembered that an additional problem in the field of law is the natural language of legal documents and the different legal provisions in individual countries.

C. Dozier et al. [17] were one of the first to conduct research based on Named Entity Recognition in the legal field. Using the example of American case law, testimonies, pleadings and other legal documents, the authors analyzed the NER. C. Cardellino et al. [18] have developed a tool to identify, classify and link legal NE. The authors focused on four different levels of detail, one of which was NER. They used a Stanford NER [19] support vector machine and a neural network in the learning algorithm. Elena Leitner et al. [20] presented the problem of fine-grained recognition of entities in legal docu-

ments. The authors developed a data set consisting of decisions of German courts, in which the source texts were manually annotated with 19 semantic classes. Then all classes were automatically generalized to seven classes (person, location, organization, legal norm, individual regulation, court order and legal literature). The results obtained show that there is no universal model with the best understanding of all classes.

II. RESEARCH METHODOLOGY

PII (Personally Identifiable Information) data detection algorithms have to be fine-tuned for each case. It is virtually impossible to obtain a high accuracy with low noise on the same pass of detection. Whereas, in the case of data related to digits, even though a number is a valid KRS/NIP (tax identification numbers) value, only semantic context might indicate whether it is a true PII or just a coincidence.

Therefore, in the case of the analysis of specific documents and adapted to a specific language (legal documents prepared in Polish), we proposed to use a combination of this methods:

- Dictionary search. A relatively easy search withing the known dictionary. The example of this method is used in our experiments where searching for first names (both female and male). The names were taken from official PESEL database³. However convenient, this list has a lot of potential to give many false positives, as the PESEL number (unique person identification number) was given to a lot of non-Polish citizens; often with short names being homonyms of common Polish words (like "Na").
- Decision rules and patterns. Using this approach, one is using regular expressions to describe the patterns of potential PII data. For instance, a NIP number is usually consisted of 10 digits divided by hyphens.
- Special type of validation for self-checking numbers. NIP number are self-checking numbers, as they use a Luhn algorithm [21], as well as additional constrains for numbers being valid. Checking this constrains is a great way to decrease greatly the number of false positives.
- Validation using external services is often used for IBAN numbers or KRS numbers in Poland. KRS number is a 10 digit number with no additional constrains, being just a next element in a single sequence of all entities in Poland. Thus, an external validator is used to make sure that the 10 digit number is an actual and up to date KRS for a valid company.
- Heuristic search using some known patterns like "sp z o.o." for a private limited company. Due to law regulations in Poland, some equities need to include mandatory information in company name.

III. COMPUTATIONAL EXPERIMENTS

The research objective of this paper was to investigate the applicability of decision rule and pattern search methods in legal document anonymisation. Therefore, in order to test the

³<https://dane.gov.pl/pl/dataset/1667,lista-imion-wystepujacych-w-rejestrze-pesel-osoby-zyjace>

proposed approach, we conducted two types of experimental studies. In the first one, we wanted to test the efficiency of the proposed solution for large amounts of data – artificial data generated for this purpose. In the second one, we performed the actual anonymisation of legal documents, and their evaluation was done by domain experts.

The first research attempt, related to achieving the stated goal of anonymising selected data in legal documents, was to test the algorithm on artificial data. In this case we used Fake Name Generator⁴ service and generated 50000 fake identities with names, company names, addresses and other sensitive information. Each line in this set should contain exactly one of name, email, phone and company name. The perfect scenario will have a mean value of detection equal to 1 for each column. The data is adapted to the Polish language and contains on average 73 words and 422 characters per line. The data preparation was followed by the application of the proposed solution to each case. The results of the experiments are presented by a box plot (Fig. 1) in which, for each PII, the minimum, 1st, 2nd (median) and 3rd quantile and maximum values are given – often these values are repeated, which is why whole box plots do not appear in the figure.

As it can be seen, in the case of the name, too many values are detected (the maximum found is 8, the median is 2, and the expected value, which is 1, is also the 1st quantile). This is because the first name often appears in the address (e.g. street name established by a famous person); in one line, in addition to the correctly detected name, the one from the address is also recognised. The ideal situation is in the case of e-mail address, where all cases were correctly detected; so was the case of phone number (there, in one case, apart from the correct number, also the false positive is detected). For the KRS number, as it is just a sequential number, false positives were found in different parts of data, as it was the case in first name. In addition to the correctly detected KRS number also NIP number is included – this explains the very low detectability of NIP number. In these experiments, once detected, the element could not be recognised again. We proposed a different approach in later experiments on real data.

The worst situation appears for the company name, which was basically not detected. This is because in our heuristic approach we require the presence of certain keywords. The generator in question was not characterised by the presence of appropriate prefixes in the name. However, in the case of legal documents, such full names are always used.

In order to assess the achievement of the final goal, the proposed method has been applied to a real-life study of a law firm operating in the territory of Poland. Fifteen legal documents written in Polish and concerning different legal scopes and cases were analysed. All documents were originally saved in DOCX format, from which, in order to apply the model, they were converted into text format. The analysis was based on the documents of different length and number of words per line (see Tab. I). It can be observed that the low

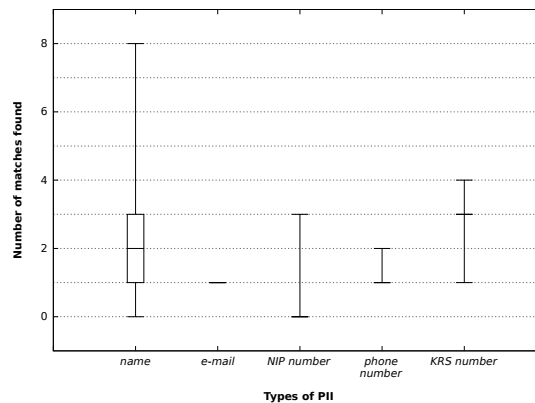


Fig. 1. Minimum, maximum, and quantiles for the prepared data

TABLE I
FEATURES OF THE DOCUMENTS ANALYSED

Legal document	# lines	# words	$\frac{\# \text{ words}}{\# \text{ lines}}$
x_1	104	500	4.81
x_2	500	3730	7.46
x_3	121	617	5.10
x_4	335	1800	5.37
x_5	162	1788	11.04
x_6	664	4291	6.46
x_7	59	294	4.98
x_8	399	2906	7.28
x_9	50	130	2.60
x_{10}	57	201	3.53
x_{11}	180	532	2.96
x_{12}	30	74	2.47
x_{13}	139	840	6.04
x_{14}	29	123	4.24
x_{15}	135	1474	10.92
sum	2964	19300	6.51
avg.	198	1287	6.50
median	135	617	4.57

number of words per line is related to the short information provided in the legal document, such as the invoice number or the service name. Therefore, the table I presents the average number of words per line.

In the next step, the fields indicated for anonymisation (with the reason for anonymising an element) were verified by specialists. They verified the PII described in Section II and each time described three known elements of the classification quality assessment:

- TP – true positives, i.e. elements that really should be anonymised for the reason given;
- FP – false positive, i.e. elements for which the indicated reason for anonymisation is not appropriate.
- FN – false negative, i.e. items that should have been anonymised for a given reason but were not.

The evaluation of TN (true negative), i.e. elements that should not be anonymised, was omitted and indeed this was not done. TN was not evaluated because, naturally, all non-annotated words mean true negative for this problem. De facto this is the number of all words in the document, except those in TP, FP and FN.

Analysing the exact anonymisation result given in Tab. II

⁴<https://www.fakenamegenerator.com>

TABLE II
RESULTS OF THE ANONYMISATION OF LEGAL DOCUMENTS

Legal document	name			e-mail			NIP number			phone number			KRS number			company name		
	TP	FP	FN	TP	FP	FN	TP	FP	FN	TP	FP	FN	TP	FP	FN	TP	FP	FN
x_1	5	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1
x_2	8	10	5	0	0	0	0	0	0	0	0	0	0	0	2	13	0	0
x_3	12	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
x_4	0	3	0	0	0	0	3	0	0	0	3	0	3	3	0	5	0	0
x_5	2	1	0	0	0	0	0	0	0	0	0	0	0	1	0	7	10	0
x_6	3	6	0	3	0	1	3	0	0	0	0	0	4	4	0	6	0	0
x_7	0	1	0	0	0	0	3	0	0	0	0	0	4	3	0	4	1	0
x_8	13	2	0	0	0	0	3	0	0	0	0	0	5	5	0	5	1	0
x_9	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
x_{10}	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2
x_{11}	4	0	1	0	0	0	6	0	0	0	0	0	3	6	0	10	1	3
x_{12}	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
x_{13}	1	0	2	0	0	0	2	0	0	0	0	0	2	2	0	2	0	0
x_{14}	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	2	0	0
x_{15}	2	0	1	0	0	0	2	0	0	0	0	0	1	2	0	1	0	1

and reading the actual document, it can be seen that in the case of PII-name there are many FP, i.e. false positive indications for anonymisation. This is mainly due to street names (consistent with first name) and first names that are also names of months (e.g. 'Maja'). FN-related errors, on the other hand, occur in the case of an atypical linguistic variant of a name.

E-mail addresses only appeared in one document and were mostly well recognised. The error associated with the address not being found was due to a transcription error (the domain extension was single character). Thus, future consideration could be given to analysing email addresses not only according to the rules, but also assuming human error in the transcription.

The situation looks very good in the case of PII-NIP number, which is a tax identification number that was correctly recognised in all documents. This is largely made possible by the Luhn algorithm. The phone number, on the other hand, did not actually appear in any document, although three times the algorithm incorrectly indicated that the found string of digits was a phone number.

An interesting situation concerns the PII-KRS number, i.e. the national court register number. All KRS numbers were detected (true positive – TP), but also other numbers in the document were incorrectly indicated as KRS. This was most often the case with the NIP number, but there were also other strings of numbers which did not relate to the KRS (it should be noted, however, that in further work, each of the elements indicated in the experiments as KRS should be anonymised anyway – whether as NIP number or national identifier of a person).

Big problems arise in the case of company names. For the analysed data, the algorithm often incorrectly indicated data for anonymisation although it was not necessary (false positive – FP). There were also situations where the company name was not detected at all. However, to sum up the work of the algorithm, in the vast majority of the cases, the algorithm detected company names that should be anonymised.

At the same time, it should be noted that in this type of documents it is a much bigger mistake to omit an element to

TABLE III
RESULTS OF THE ANONYMISATION OF LEGAL DOCUMENTS IN TERMS OF TP, FP, FN AND IN TERMS OF MEASURES TO ASSESS CLASSIFICATION QUALITY: PRECISION, RECALL AND F-SCORE

Legal document	TP	FP	FN	precision	recall	F-score
x_1	6	1	1	0.8571	0.8571	0.8571
x_2	10	23	5	0.3030	0.6667	0.4167
x_3	12	1	1	0.9231	0.9231	0.9231
x_4	11	6	0	0.6471	1.0000	0.7857
x_5	9	12	0	0.4286	1.0000	0.6000
x_6	19	6	1	0.7600	0.9500	0.8444
x_7	11	2	0	0.8462	1.0000	0.9167
x_8	26	3	0	0.8966	1.0000	0.9455
x_9	8	0	0	1.0000	1.0000	1.0000
x_{10}	2	2	2	0.5000	0.5000	0.5000
x_{11}	23	1	4	0.9583	0.8519	0.9020
x_{12}	3	0	0	1.0000	1.0000	1.0000
x_{13}	7	0	2	1.0000	0.7778	0.8750
x_{14}	4	0	0	1.0000	1.0000	1.0000
x_{15}	6	0	2	1.0000	0.7500	0.8571
sum	157	57	18	0.7336	0.8971	0.8072
avg.	10	4	1	0.8080	0.8851	0.8448
median	9	1	1	0.8966	0.9500	0.9225

be anonymised (i.e. those indicated in FN) than to anonymise it incorrectly (in our case FP). In addition, often elements that showed FP for one of the methods were in fact anonymised anyway for another reason. Therefore, in Tab. III, the summary results are presented, in which the reason for anonymisation is not indicated, but only the information whether a given word should be anonymised or not.

With these values, it can be indicated that in all analysed documents 185 words should have been anonymised (out of a total of 19300 words), with 168 words actually detected, 17 words not anonymised (although they should have been), and 57 words incorrectly indicated as requiring anonymisation. This gives a median of 10 words per document (requiring anonymisation) while omitting 1 word per document and indicating 1 additional word incorrectly. This allows us to assess the effectiveness of the solution.

With the values in Tab III, it is possible to calculate measures of classification evaluation, such as *precision* (eq. (1)), *recall* (eq. (2)) and *F-score* (eq. (3)). These are important

measures, where the *precision* determines with what certainty we can trust the classifier that a given element (word) should actually be anonymised, while *recall* determines how many elements (words) that should be anonymised in the document have been indicated as anonymised, and *F – score* is the harmonic mean of *precision* and *recall*.

$$precision = \frac{TP}{TP + FP} \quad (1)$$

$$recall = \frac{TP}{TP + FN} \quad (2)$$

$$F - score = \frac{TP}{TP + 0.5 \cdot (FP + FN)} \quad (3)$$

As mentioned earlier, from the point of view of the anonymisation of legal documents, it is more important to detect the elements that should be anonymised. Therefore, when analysing the results, the recall measure should be optimised first. The results related to the quality assessment of the anonymisation work are presented in Tab. III. In the case of the proposed solution, our method achieves a recall of 89.71% (in terms of the sum of all documents) and 95.00% in terms of the median recall calculated separately for each legal document.

Of course, it is important to note that all the results reported in this section relate to the analysis of only 6 types of PII: name, e-mail address, tax identification number (NIP number), phone number, national court register number (KRS number) and company name. Other data that should actually be anonymised have not been analysed by us at this stage of the project – the proposed solution. For subsequent work, a more sophisticated use machine learning and natural language processing methods is required.

IV. CONCLUSIONS AND FUTURE WORKS

It should be emphasised that the research objective was to analyse a limited range of data – only a selected type of data was anonymised. Thus, we wanted to assess the applicability of IT techniques and decision rules and patterns in document anonymisation. The conducted experiments confirm that the solutions we have applied allow us to obtain good results – this is particularly evident in the case of real legal documents. Therefore, in the future, in addition to improving the rules proposed so far, we believe that it is worth to develop the modelling of machine learning on bigger groups of real legal documents that can be used as learning data in supervised machine learning models.

Future work may include developing a multi-criteria model for multi-step analysis. It is possible to initially verify data on the basis of unit names, and then carry out a thorough analysis on the basis of the content only for selected data. Such a solution is possible with the use of machine learning methods and natural language processing techniques. However, it should be remembered that the task of creating a universal system that would recognize all classes is very difficult, and

the effectiveness of anonymisation largely depends on the structure of the documents that could be referred to given types of legal documents.

REFERENCES

- [1] P. Štarchoň and T. Pikulík, “Gdpr principles in data protection encourage pseudonymization through most popular and full-personalized devices-mobile phones,” *Procedia Computer Science*, vol. 151, pp. 303–312, 2019.
- [2] M. Mozes and B. Kleinberg, “No intruder, no validity: Evaluation criteria for privacy-preserving text anonymization,” *arXiv preprint arXiv:2103.09263*, 2021.
- [3] P. Regulation, “Regulation (eu) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/ec (general data protection regulation) (text with eea relevance),” *Regulation (eu)*, vol. 679, p. 2016, 2016.
- [4] I. Glaser, T. Schamberger, and F. Matthes, “Anonymization of german legal court rulings,” in *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Law*, 2021, pp. 205–209.
- [5] G. M. Csányi, D. Nagy, R. Vági, J. P. Vadász, and T. Orosz, “Challenges and open problems of legal document anonymization,” *Symmetry*, vol. 13, no. 8, p. 1490, 2021.
- [6] B. Mohit, “Named entity recognition,” in *Natural language processing of semitic languages*. Springer, 2014, pp. 221–245.
- [7] T. H. Cao, T. M. Tang, and C. K. Chau, “Text clustering with named entities: a model, experimentation and realization,” in *Data mining: Foundations and intelligent paradigms*. Springer, 2012, pp. 267–287.
- [8] R. Grishman and B. M. Sundheim, “Message understanding conference-6: A brief history,” in *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*, 1996.
- [9] H. Vico and D. Calegari, “Software architecture for document anonymization,” *Electronic Notes in Theoretical Computer Science*, vol. 314, pp. 83–100, 2015.
- [10] B. M. Sundheim, “Overview of results of the muc-6 evaluation,” in *Sixth Message Understanding Conference (MUC-6): Proceedings of a Conference Held in Columbia, Maryland, November 6-8, 1995*, 1995.
- [11] D. Nadeau and S. Sekine, “A survey of named entity recognition and classification,” *Linguisticae Investigationes*, vol. 30, no. 1, pp. 3–26, 2007.
- [12] O. Kabasakal and A. Mutlu, “Named entity recognition in turkish bank documents,” *Kocaeli Journal of Science and Engineering*, vol. 4, no. 2, pp. 86–92, 2021.
- [13] F. Graliński, K. Jassem, M. Marcińczuk, and P. Wawrzyniak, “Named entity recognition in machine anonymization,” *Recent Advances in Intelligent Information Systems*, pp. 247–260, 2009.
- [14] J. Piskorski, “Named-entity recognition for polish with sprout,” in *Intelligent Media Technology for Communicative Intelligence*. Springer, 2004, pp. 122–133.
- [15] B. Kleinberg and M. Mozes, “Web-based text anonymization with node.js: Introducing netanos (named entity-based text anonymization for open science),” *Journal of Open Source Software*, vol. 2, no. 14, p. 293, 2017.
- [16] D. Reynders, “Digitalising justice systems to bring out the best in justice,” *Eucriim: the European Criminal Law Associations’ forum*, no. 4, pp. 236–237, 2021.
- [17] C. Dozier, R. Kondadadi, M. Light, A. Vachher, S. Veeramachaneni, and R. Wudali, “Named entity recognition and resolution in legal text,” in *Semantic Processing of Legal Texts*. Springer, 2010, pp. 27–43.
- [18] C. Cardellino, M. Teruel, L. A. Alemany, and S. Villata, “A low-cost, high-coverage legal named entity recognizer, classifier and linker,” in *Proceedings of the 16th edition of the International Conference on Artificial Intelligence and Law*, 2017, pp. 9–18.
- [19] J. R. Finkel, T. Grenager, and C. D. Manning, “Incorporating non-local information into information extraction systems by gibbs sampling,” in *Proceedings of the 43rd annual meeting of the association for computational linguistics (ACL’05)*, 2005, pp. 363–370.
- [20] E. Leitner, G. Rehm, and J. Moreno-Schneider, “Fine-grained named entity recognition in legal documents,” in *International Conference on Semantic Systems*. Springer, 2019, pp. 272–287.
- [21] H. P. Luhn, “Computer for verifying numbers,” *US Patent*, vol. 2, no. 950, p. 048, 1960.

Performance Management of IT Professionals: A Humanistic Model

Marcus Vinicius Alencar Terra
State University of Londrina
Computing Department
P.O. Box 10.011
Londrina, Paraná, Brazil
Postal Code: 86.057-970
Email: marcusterra@gmail.com

Vanessa Tavares de Oliveira Barros
State University of Londrina
Computing Department
P.O. Box: 10.011
Londrina, Paraná, Brazil
Postal Code: 86.057-970
Email: vanessa@uel.br

Rodolfo Miranda de Barros
State University of Londrina
Computing Department
P.O. Box: 10.011
Londrina, Paraná, Brazil
Postal Code: 86.057-970
Email: rodolfo@uel.br

Abstract—Nowadays society has transformed performance-related issues into situations that are merely focused on goals and competitiveness, which generates, in IT professionals, the feeling of being under constant pressure due to the need for immediate delivery of results. For this reason, Human Performance Management has become an increasingly essential process inside organizations, through which it is possible to improve, among other things, efficiency, productivity and satisfaction of employees. Its benefits are even more evident in areas such as Information Technology (IT), where evolution, complexity and uncertainty are ever-present factors. Based on this scenario, this study proposes a performance management model for IT professionals, linked to the philosophical current of Humanism and addressing aspects such as equality, respect, participation, competency and personal development. Thus, this research intends to have a positive effect on IT performance (individual/team/organizational), humanizing the relationship between IT employees and their organizations.

Index Terms—performance management, performance appraisal, IT professional, humanism, model

I. INTRODUCTION

HUMAN performance management (HPM) is an essential process inside organizations. The main objective of this type of management is to provide a performance measure for activities carried out by employees, while promotes the improvement of productivity, motivation and satisfaction. Therefore, this process can be considered a structuring basis for developing organizational culture and relationship between organization and its employees [1][2].

The benefits of an effective performance management are even more evident in areas such as Information Technology (IT) where complexity, dynamism and innovation are ever-present factors, requiring professionals to have high levels of knowledge and creativity [3][4][5]. Thus, the main motivation for this study comes from the opportunity to develop an analysis regarding performance management and satisfaction of IT professionals inside organizations, proposing, as a result, a management model composed by a set of guidelines based on the principles of Humanism.

Evaluate IT professionals performance and define the skills needed for the job are not recent concerns [6] [7] [8] [9] [10], however, nowadays society, as a result of economic globalization, has transformed issues related to performance

into situations merely focused on goals and competitiveness, which generates, on IT employees, the feeling of being under constant pressure due to the need for immediate delivery of results.

The high turnover of professionals in Information Technology area [11][12][13] and occupational diseases, like technostress [14], are just some of the problems related to performance management, specially, when it is primarily focused on impersonal behavior, productivity and value delivery.

Given this scenario, several initiatives have emerged seeking to humanize and improve the relationship between employees and organizations [15][16][17][18][19]. Following the same line of thought, the purpose of this research is to define a performance management model for IT professionals linked to the philosophical stance of Humanism, considering, therefore, the issues related to a human centered management and addressing aspects of human nature, such as dignity, limits, aspirations, capabilities and potential.

Other important contributions of this study are focused on: detailing organizational culture concepts and human performance management; humanistic ethics' analysis in corporate environment; knowledge structuring for developing more effective methods of appraisal. Such contributions aims to be adherent to the current and future reality of IT professionals.

This research is believed to be scientifically original, since it proposes, as far as is known, a unique model for human performance management, based on the perspective of IT profile singularities together with extremely important concepts for society, like organizational culture, humanism and ethics.

It also can be justified by the proposition of a management model potentially capable of improve people and organizations, representing an extremely relevant artifact in a context full of uncertainties, challenges and constant transformations. In addition, the results obtained by this research are expected to server as groundwork for future studies focused on producing new knowledge, frameworks and methodologies related to HPM.

The rest of this article is structured as follows: section 2 sets out the theoretical foundation and related works considered for the research; section 3 exposes the context and issues involved

in this study; section 4 describes the scientific methodology employed; section 5 describes the proposed solution; section 6 analyzes and discusses the obtained results; and, finally, section 7 presents the last considerations of the research.

II. LITERATURE REVIEW

A. Organizational Culture Theory

In order to analyze employee performance, engagement and satisfaction in the context of organizations, it is necessary to understand organizational culture and its influence on these factors. [20].

The most commonly used and accepted definition of organizational culture is the one proposed by Schein (1988, p. 7) [21][20][22]:

A pattern of basic assumptions, invented, discovered, or developed by a given group, as it learns to cope with its problems of external adaptation and internal integration, that has worked well enough to be considered valid and, therefore is to be taught to new members as the correct way to perceive, think, and feel in relation to those problems.

As part of the Organizational Culture Theory, Schein (1988) [21] also proposes to describe an organization's culture as a set of levels that represent its elements, as shown in Figure 1.

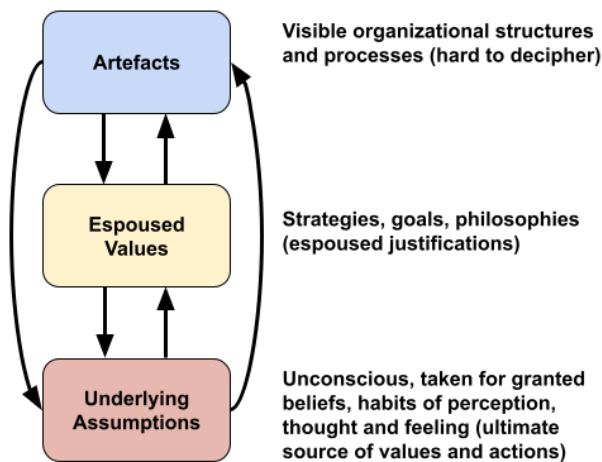


Fig. 1. The Levels of Organizational Culture. Adapted from [21]

In the context of human performance, **Artefacts** represent what people effectively accomplish and which directly reflect on performance and goal achievement. This level is the most evident in the culture, although some organizational actions and structures can be difficult to understand and justify.

The **Espoused Values** are the strategies, objectives, norms and philosophies openly propagated by the organization, aiming, for example, the effectiveness of the performance management. The organization's actions have the most significant effect at this cultural level.

Finally, **Underlying Assumptions** are taken-for-granted beliefs based on thoughts, perceptions, and feelings. At this level,

cultural elements become visible only through the analysis of behavioral patterns. These underlying assumptions exist in order to simplify complex issues of organizational reality, such as the reasons why one person is promoted over another [23].

It is evident that the elements of each level of organizational culture are capable of influencing all the others, however, underlying assumptions of an organization have a great impact on its artifacts, supplanting, in some situations, the influence of declared values [21][23].

Organizational culture can be seen as a preponderant factor in performance management but also as part of the results of this management, that is, by promoting the performance improvement of employees and their teams, organizational culture ends up being directly influenced by this improvement [22].

Regarding the performance of an organization, it is possible to build a culture that emphasizes essential points, such as meritocracy, transparency and recognition [24]. All these positive changes that take place in the organizational culture have the power to provide employees with the possibility to act proactively, identifying, mitigating and eliminating human errors and correcting the organization's vices and weaknesses [25].

A favorable cultural posture results in actions that can, if carried out accordingly, guide the institution towards an effective management of human performance. [25].

B. Human Performance Management

Interest in the effectiveness of Human Resource Management, or its more recent term *Human Management* [26], became more evident and frequent from the 1970s onward [2].

Human Performance Management is one of the main pillars of Human Resource Management and, for this reason, the study of human performance inside organizations also started more than half a century ago [6][7][8][9][10]. Aspects related to this branch of organizational management have, therefore, a rich literature that can be found in the most varied areas of research, such as psychology, administration, sociology, information systems and economics [27][28].

In addition to the Organizational Culture Theory, HPM is based on a wide range of other theories [29][30][31], which demonstrates the high complexity and deepness of the subject.

The Table I presents some fundamental theories to understand and develop Human Performance Management.

Based on the numerous definitions of HPM that can be found in the literature, it can be inferred that Human Performance Management is a cyclical and continuous process that is intended to identify, plan, measure, control and develop performance at work, both individually and as a team, while aligning this performance with the organization's strategic objectives and the value delivery from executed activities [32][26][33][34][35].

HPM is, therefore, a complex process that involves a series of methodologies, techniques and approaches focused on overcoming the challenges and difficulties inherent to this type of management and returning positive results for organizations [36]. Figure 2 presents a holistic and pragmatic

TABLE I
FUNDAMENTAL THEORIES OF HUMAN PERFORMANCE MANAGEMENT [29][30][31]

Theory	Author(s)	Theory	Author(s)
Theory of Action and Job Performance	Richard E. Boyatzis	Agency Theory	Michael C. Jensen and William H. Meckling
Attribution Theory	Fritz Heider	Bureaucratic Theory	Max Weber
Field Theory	Kurt Lewin	Competency Theory	David McClelland
Contract Theory	Oliver Hart and Bengt Holmström	Theory of Individual Differences in Task and Contextual Performance	Stephan J. Motowidlo, Walter C. Borman, and Mark J. Schmit
Two-Factor Theory	Frederick Herzberg	Equity Theory	J. Stacy Adams
Expectancy Theory	Victor Vroom, Lyman Porter and Edward Lawler	Goal-setting Theory	Edwin Locke and Gary Latham
General Systems Theory	Ludwig von Bertalanffy	Organizational Justice Theory	Jerald Greenberg
Theory of Behavioral Engineering Model	Thomas F. Gilbert	Theory of Human Motivation	Abraham Maslow
Achievement Motivation Theory	David McClelland	Theory of the Social Self	George H. Mead
Job Characteristics Theory	Richard Hackman, Edward Lawler, and Gred Oldham	Bases of Social Power Theory	John French and Bertran Raven
Reinforcement Theory	B. F. Skinner	Theory of Social Exchange	George C. Homans

view of the Human Performance Management Framework as proposed by [32].

C. Humanism

Humanism is essentially a philosophical stance that assigns preeminent importance to human beings, their experiences, interests and rights. The hallmark of humanist philosophy is, therefore, the development of people's potential, considering Protagoras' relativism (490-420 BCE) where "man is the measure of all things" [37].

Among all the principles contemplated by humanism, some of them deserve to be highlighted: human value; individual dignity; the pursuit of civic culture; promotion of diversity and equality; and humanistic ethics. [37].

According to humanist ethics, the human being must "be considered as an end and never exclusively as a means or instrument for any purpose external to itself" [38]. Thus, moral rules are defined from the perspective of humanity, that is, *right* is everything that is good for human beings, values their life and develops their capacities, while *wrong* is everything that harms or takes away human dignity, represses individuality and dehumanizes people [38].

One of the main global aspirations of Humanism is found in the Universal Declaration of Human Rights, which establishes a commitment to promote universal respect for and observance of human rights and fundamental freedoms, demonstrating that human beings and their dignity must be above private power in any sphere [39].

Based on the fundamental idea of humanism, many other reflections have been developed, also covering the organizational context [40]. Thus, inside organizations, humanist management must place human dignity and rights as central concerns in all its subjects and methodologies [41]. In this sense, economic transactions are considered, in essence, as relationships between people and, for this reason, organizations need to serve the objectives of humanity and not the opposite. In doing so, people are seen as active and central elements of the economic system and not passive and secondary objects of an economy guided by other goals [41].

In a concise manner, a humanistic management is concerned with human needs and oriented towards the complete and extensive development of human being virtues. [42].

Thus, based on the concepts of this type of management, it is possible to describe a progressive model of 3 levels of entrepreneurial humanism [19][43], as shown in figure 3. In the same sense, humanistic management is composed of 5 dimensions [40]:

- Managerial responsibility
- Employee motivation
- Personal promotion
- Interpersonal relationships
- Organizational culture

Regarding the relationship between technology and humanism, it is important to note that this is not a recent issue [44], but the advancement of Artificial Intelligence and its application in real Information System problems has leveraged new initiatives seeking to discuss this relationship [45][46]. Such efforts are known as Digital Humanism, which has proclaimed and disclosed a manifesto with principles on current and future technological development, as well as, on the co-evolution of technology and humanity [47][46].

D. IT Professional Profile

There is a vast literature devoted to the study of professionals working in the area of Information Technology, there are many documents that analyze or propose aspects related to the profile of this type of profession. Authors from the 60s and 70s were already concerned with the topic and proposed ways of defining and evaluating the fundamental characteristics of these professionals [9][7][6][8][10][48]. On the other hand, more recent studies point to an extremely complex and plural profile, capable of acting in different areas of the organization [49][50][51][52][18].

For this reason, precisely defining the IT professional and his/her profile has become a controversial task, where the conclusion can even be that everyone in the organization is somehow part of the workforce that works in IT area [18]. Despite this, this study proposes and uses a simple and broad

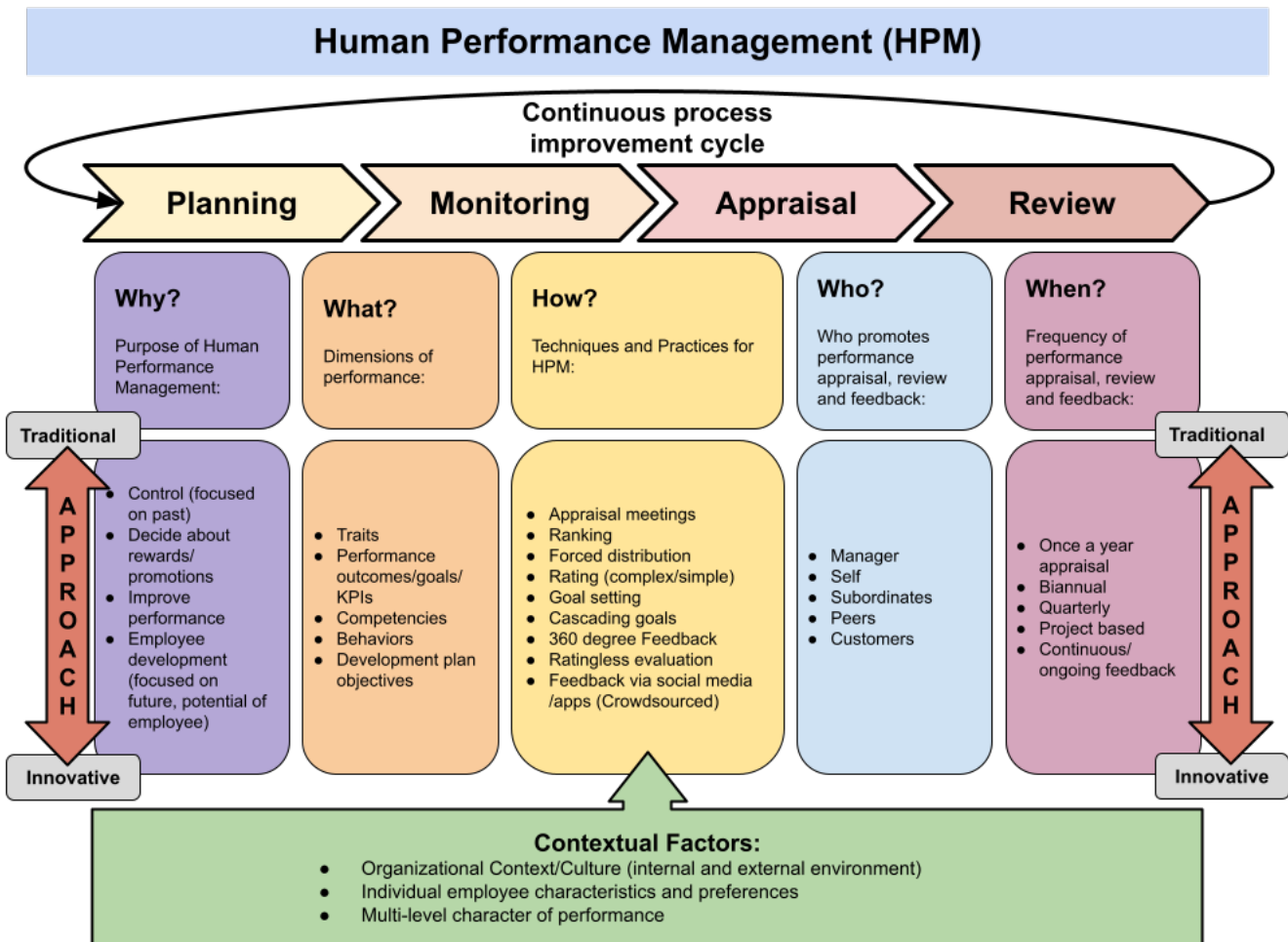


Fig. 2. Human Performance Management Framework. Adapted from [32]



Fig. 3. Levels of Entrepreneurial Humanism. Based on [19].

definition of IT professionals that summarizes the mission of these workers [53][54]:

Professionals who have attributions and perform activities, primarily, aimed at delivering information technology products and services with effectiveness and security.

Based on this definition, this research considered that any professional related directly to Information Technology fields, such as Information Systems (IS), Communications (ICT) and Data Science, is an IT professional.

Even with the diversity of characteristics and classifications pertinent to IT profile, it is possible to identify convergent aspects common to all professionals from this area [50]. The IT employee has unique identity, knowledge, skills, attitudes and interests, his/her focus at work tends to be more centered on technical issues, to the detriment of interpersonal skills such as communication [55]. When compared to professionals from other areas, IT personnel have a greater desire for opportunities, challenges and autonomy [56], in addition, they are motivated by achievements, recognition, constant learning and personal growth [55].

These are also common elements of the IT area [50]:

- Supply and demand for IT professionals continually changing.
- Requirement of a combined set of technical, humanistic and business skills.
- Professional environment in constant change, requiring adaptation and update of specific skills in short periods of time.
- Activities developed are essentially cognitive and difficult to monitor and evaluate.

As mentioned before, the set of required IT skills can be classified into 3 broad categories [50][56]:

- **Technical Skills:** knowledge and competences related to the use and application of technologies.
- **Humanistic Skills:** temperance, resilience, interpersonal relationships (e.g., teamwork, leadership, communication), promotion of well-being.
- **Business Skills:** business domain knowledge, project management, ethics, problem and conflict resolution.

The definitions and all other considerations just presented are what make Information Technology professionals singular inside organizations, demanding differentiated attention and approach from human resource management and, more specifically, from human performance management. [57][55][56][50][54].

III. CONTEXT AND PROBLEM

As demonstrated in Figure 2, Human Performance Management is a complex process that involves numerous variables, methodologies, people and applications. In addition, HPM is based on a vast amount of theories, in the most diverse areas of research, which makes the study on the subject very extensive and deep.

Considering the theories studied, the Organizational Culture theory proved to be the central point of analysis, through which it is possible to understand how the actions defined in the HPM are actually performed and the effects that this management has on employees.

Added to this, there are the unique characteristics of IT professionals and their work environment, which require, from Human Performance Management, a differentiated and individualized treatment for employees working in this area.

The particularities, problems and difficulties faced by Information Technology professionals have been subject of study for many years [51], considering the issues addressed by the researchers, the following items deserve to be highlighted:

- Ethics in IT and Conflicts at work.
- Job or career turnover.
- Gender imbalance/prejudice in IT.
- Treatment of minorities in IT.
- Evolution/Change of project and work models.
- Overload, stress, exhaustion and burnout.
- Work versus social life conflicts.
- Speed of technological evolution.
- Perception of professional stagnation or obsolescence.

It is important to understand that an effective HPM has the power to treat, mitigate or even solve the problems experienced in IT area.

In addition, the philosophical thought of Max Weber (1864-1920) states that work is a source of dignity and nobility for human existence, promoting life and well-being, for oneself and for others [58]. Based on this vision, it is essential to study, develop and encourage a more humane and sustainable organizational management.

In this sense, Teehanke (2021)[43] proposes that human beings should have the following needs considered and managed by organizations:

- Physical and mental health.
- Intellectual development.
- Emotional growth.
- Experiences in the fields of arts, culture and aesthetics.
- Social connectivity.
- Moral and spiritual development.

In order to ensure the true evolution of human performance, HPM must consider the humanistic aspects in its management premises. Thus, it is essential to leave behind the model where workers are seen only as production resources, which need to be monitored and optimized considering only productivity and efficiency [43].

IV. METHODS

In order to develop the solution proposed in this research, the *Design Science Research* paradigm was used [59]. Based on this paradigm, the first 4 steps of the process model proposed by Peffers et al. (2007) [60], known as *Design Science Research Methodology* (DSRM), were carried out, as shown in figure 4.

Thus, this research, which has a qualitative approach and positivist epistemological position, was developed in 4 phases or stages:

- 1) Problem identification and motivation.
- 2) Definition of solution objectives.
- 3) Design and development.
- 4) Demonstration.

In the first stage, an extensive literature review was performed, which made it possible to identify the context and domain of the problem. The motivation and reasons for seeking a solution were also determined, as indicated in Section III.

Continuing in the second phase, with the problem identified, it was possible to establish the objectives of the proposed solution, the assumptions and requirements that the produced artifact should follow. At this point, the need to build a human performance management model specifically focused on information technology professionals was defined, encompassing the precepts and characteristics of humanism.

In the third stage, the design and construction of the proposed model allowed a better understanding of the problem domain and its solution.

Finally, in the last step, the applicability of the model was demonstrated through evaluation methods of static and architectural analysis [59], verifying the structure of the model and studying its suitability to the requirements and theoretical assumptions postulated.

V. RESULTS

Based on the context and issues described in Section III, the present work proposes a humanistic model for performance management focused on developing the needs of IT employees as human beings, guided by the principles of ethics and appreciation of life.

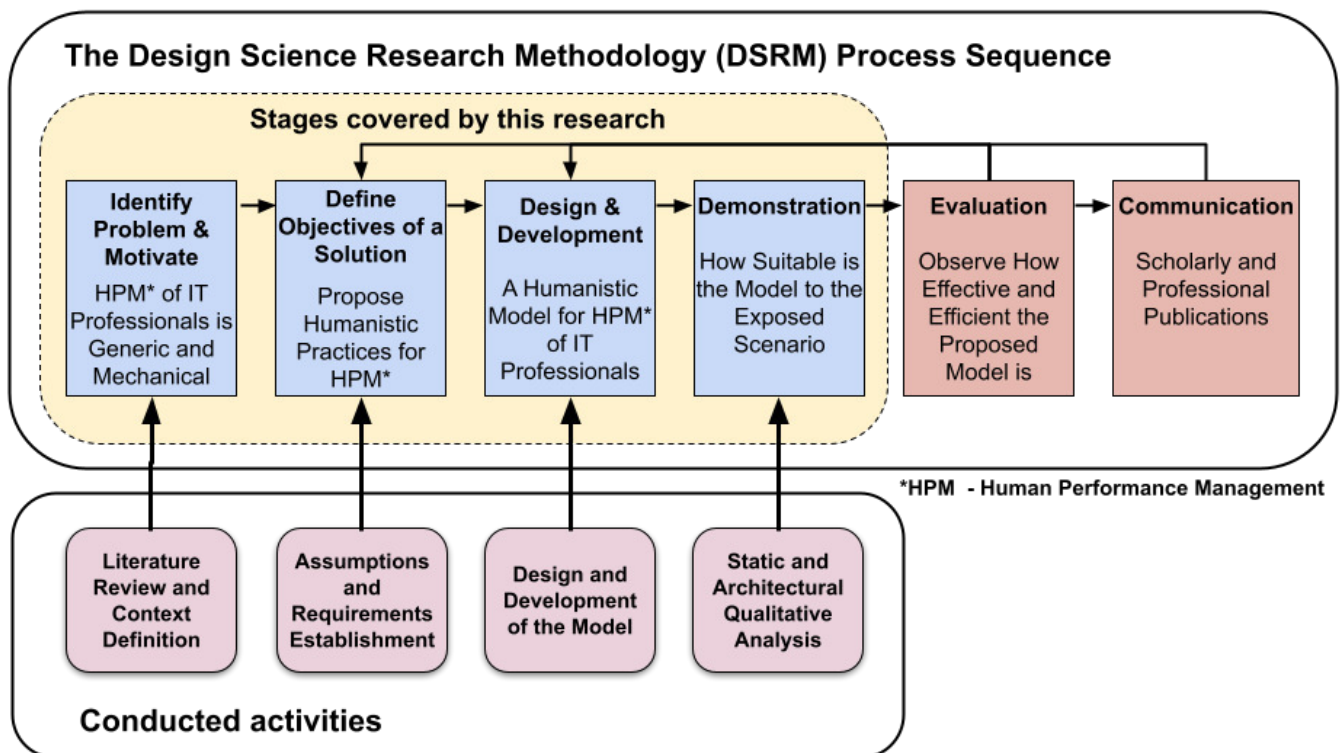


Fig. 4. Research Implementation using the DSRM Process Model. Adapted from [60]

The solution has the following fundamental assumptions and requirements:

- Promotion of the human person: all the steps on the process must preserve and improve IT employees' dignity and expectations;
- Ethics: all tasks related do performance evaluation need to be based on ethics principals defined by society and organizational culture;
- Transparency: actions and decisions of performance management must be communicated widely and transparently;
- Justice, Equality and Inclusion: the process must pursue and guarantee organizational justice, equality and inclusion;
- Objectivity and Celerity: HPM should avoid excessive bureaucracy, prioritizing simplicity, objectivity and celerity;
- Active stakeholder participation: all the steps must permit and promote the participation of organization community;
- Formative Appraisal: performance evaluation must prioritize a formative method (perceptions) over a summative one (grades and rates);
- Respect for individuality and dignity: all persons involved in the process must be treated with respect and dignity;
- Personal and professional development: HPM must ensure the personal and professional development of IT employees.

The details of the proposed model can be seen in Figure 5, there are 4 distinct phases in the human performance

management process (planning, performance appraisal, review, monitoring and improvement), where each step is specified according to humanistic foundations.

As can be seen from the model, **human being** must be the central concern of HPM, this indicates that management success depends, primarily, on how the issues related to humans are treated.

IT Profile is the next element that deserves consideration in the model, this represents all characteristics, skills, competences, knowledge and needs that compose and differentiate IT professionals from the others. The actions taken during the performance management process must respect and consider this profile.

The **planning** stage defines and presents the individual and collective purposes for evolution of human performance of organization's members. Planning must have a person-centered approach, be public and objective, in addition, it must address the personal and professional growth needs of each individual, contemplating the specificity of the IT profile. Its realization has collaborative characteristics, where decisions about what to produce, develop, improve or remedy are taken jointly and equitably.

Performance appraisal has a formative character and occurs continuously, being encouraged and carried out by all members of the organizational community. It does not use a summative method, the absence of values prevents comparisons and, at the same time, develops empathy and critical thinking in employees when interpreting feedback received or

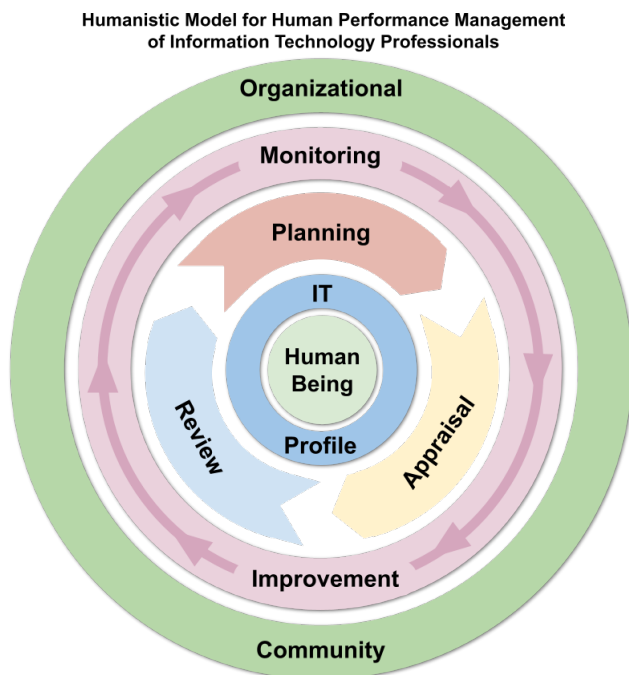


Fig. 5. Humanistic Model for Human Performance Management of Information Technology Professionals.

evaluating a colleague or themselves.

The **review** stage, which is also participatory, analyzes what has been carried out, allowing the necessary points to be adjusted and improving HPM proposals and mechanisms for the new cycle that will begin.

In the **monitoring and improvement phase (follow-up and support)**, the entire organizational community is invited to collaborate with the development of personal and professional goals, both individual and collective, which were defined in the planning stage. This step occurs throughout HPM process and can be executed in the form of mentoring, coaching, physical and mental health support, transfer and promotion of knowledge, among others possibilities. In addition, actions that seek to improve the management process itself are also promoted and encouraged.

VI. DISCUSSION

The analysis of the results relied on the evaluative precepts proposed by Design Science Research [59] and, from there, we sought to understand how adequate the solution is in relation to the human performance management framework (Figure 2), to the humanist principles and to the characteristics and particularities of the IT professional profile.

Regarding the human performance management framework (Figure 2), it is possible to verify that the proposed model is fully adherent to it. The solution presents a management process with well defined phases and elements, deals with organizations' contextual factors and allows the adoption of existing approaches and methodologies related to the field of HPM.

Considering the problem covered in this research in relation to humanist principles and the profile of the IT professional, it is understood that the following points are contemplated in the solution, as proposed by Farah (2000, p. 150) [38]:

- Respect for human beings and equal treatment.
- Improvement of Organizational Justice.
- Concern with personal (physical, mental, moral and spiritual) and professional development (skills, knowledge and technological evolution).
- Planning and evolution of human performance in an appropriate manner, linked to both, personal and organizational, expectations.
- Collaboration in IT talent retention.

This study is considered innovative because it proposes an unprecedented model capable of improving, from a humanist perspective, the human performance management of information technology professionals.

To corroborate this statement, in most of the studies on HPM analyzed by this research, the authors focused on directly applicable methods or management practices for performance evaluation, and, only in few cases, the characteristics of IT professionals are encompassed, however, leaving aside the question of humanism. The authors who dealt with humanistic management, on the other hand, do not, specifically, address performance management or the profile of the IT professional.

VII. CONCLUSIONS

For Humanism, every person is worthy of development [37]. This research was conducted based on this humanistic axiom and it is hoped that this study can somehow contribute to a better and more sustainable world, aware of the importance of valuing life and human dignity.

The limitations of the model and its usage are related to the issues concerning the initial adoption, since the study considered a generic process of HPM, for more specific scenarios or situations, it will be necessary make adaptations and develop extensions in order to deliver the expected results.

As future work, the development of complementary research is indicated, which, based on the proposed model, will produce new artifacts capable of leveraging the evolution of Human Performance Management of Information Technology professionals. Good practices, processes, methodologies, surveys and information systems are just a few examples of important artifacts that can be defined and explored.

REFERENCES

- [1] A. DelPo, *Performance Appraisal Handbook, The: Legal & Practical Rules for Managers*, ser. Performance Appraisal Handbook. NOLO, 2007. ISBN 9781413305678
- [2] G. Latham, K. N. Wexley, and K. Wexley, *Increasing Productivity Through Performance Appraisal*, ser. Addison-Wesley series on managing human resources. Addison-Wesley, 1981. ISBN 9780201042177
- [3] T. Kanij, J. Grundy, and R. Merkel, "Performance appraisal of software testers," *Information and Software Technology*, vol. 56, no. 5, pp. 495–505, 2014. doi: 10.1016/j.infsof.2013.11.002 Performance in Software Development. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0950584913002164>
- [4] L. Fernandez-Sanz, "Personal skills for computing professionals," *Computer*, vol. 42, no. 10, pp. 110–111, 2009. doi: 10.1109/MC.2009.329

- [5] B. L. Killingsworth, M. B. Hayden, D. Crawford, and R. Schellenberger, "A model for motivating and measuring quality performance in information systems staff," *Information Systems Management*, vol. 18, no. 2, pp. 8–14, 2001. doi: 10.1201/1078/43195.18.2.20010301/31271.2. [Online]. Available: <https://doi.org/10.1201/1078/43195.18.2.20010301/31271.2>
- [6] R. M. Berger and R. C. Wilson, "Correlates of programmer proficiency," in *Proceedings of the Fourth SIGCPR Conference on Computer Personnel Research*, ser. SIGCPR '66. New York, NY, USA: Association for Computing Machinery, 1966. doi: 10.1145/1142620.1142629. ISBN 9781450378109 p. 83–95. [Online]. Available: <https://doi.org/10.1145/1142620.1142629>
- [7] R. A. Dickmann, "A programmer appraisal instrument," in *Proceedings of the Second SIGCPR Conference on Computer Personnel Research*, ser. SIGCPR '64. New York, NY, USA: Association for Computing Machinery, 1964. doi: 10.1145/1142635.1142640. ISBN 9781450378116 p. 45–64. [Online]. Available: <https://doi.org/10.1145/1142635.1142640>
- [8] B. Powell, "Performance evaluation of programmers and analysts," in *Proceedings of the 3rd Annual ACM SIGUCCS Conference on User Services*, ser. SIGUCCS '75. New York, NY, USA: Association for Computing Machinery, 1975. doi: 10.1145/800115.803716. ISBN 9781450374170 p. 19–21. [Online]. Available: <https://doi.org/10.1145/800115.803716>
- [9] J. C. Hoyle and R. D. Arvey, "Development of behaviorally based rating scales," in *Proceedings of the Tenth Annual SIGCPR Conference*, ser. SIGCPR '72. New York, NY, USA: Association for Computing Machinery, 1972. doi: 10.1145/800156.805029. ISBN 9781450374620 p. 85–103. [Online]. Available: <https://doi.org/10.1145/800156.805029>
- [10] D. B. Mayer and A. W. Stalnaker, "Selection and evaluation of computer personnel- the research history of sig/cpr," in *Proceedings of the 1968 23rd ACM National Conference*, ser. ACM '68. New York, NY, USA: Association for Computing Machinery, 1968. doi: 10.1145/800186.810630. ISBN 9781450374866 p. 657–670. [Online]. Available: <https://doi.org/10.1145/800186.810630>
- [11] S. Sethunga and I. Perera, "Impact of performance rewards on employee turnover in sri lankan it industry," in *2018 Moratuwa Engineering Research Conference (MERCOn)*. Institute of Electrical and Electronics Engineers, 2018. doi: 10.1109/MERCOn.2018.8421961 pp. 114–119.
- [12] S. Renaud, L. Morin, J.-Y. Saulquin, and J. Abraham, "What are the best HRM practices for retaining experts? a longitudinal study in the canadian information technology sector," *International Journal of Manpower*, vol. 36, no. 3, pp. 416–432, jun 2015. doi: 10.1108/ijm-03-2014-0078. [Online]. Available: <https://doi.org/10.1108/ijm-03-2014-0078>
- [13] M. Riemenschneider, Cynthia; Allen and M. Reid, "'potential antecedents to the voluntary turnover intentions of women working in information technology,'" in *Proceedings of 2002 Americas Conference on Information Systems (AMCIS)*. Association for Information Systems, 2002, pp. 2018–2022. [Online]. Available: <https://aisel.aisnet.org/amcis2002/277>
- [14] C. Maier, S. Laumer, J. Wirth, and T. Weitzel, "Technostress and the hierarchical levels of personality: a two-wave study with multiple data samples," *European Journal of Information Systems*, vol. 28, no. 5, pp. 496–522, 2019. doi: 10.1080/0960085X.2019.1614739. [Online]. Available: <https://doi.org/10.1080/0960085X.2019.1614739>
- [15] B. Gaur, "Hr4.0: An analytics framework to redefine employee engagement in the fourth industrial revolution," in *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*. Institute of Electrical and Electronics Engineers, 2020. doi: 10.1109/ICCCNT49239.2020.9225456 pp. 1–6.
- [16] K. Idell, D. Gefen, and A. Ragowsky, "Managing it professional turnover," *Commun. ACM*, vol. 64, no. 9, p. 72–77, aug 2021. doi: 10.1145/3434641. [Online]. Available: <https://doi.org/10.1145/3434641>
- [17] K. Moon, "Specificity of performance appraisal feedback, trust in manager, and job attitudes: A serial mediation model," *Social Behavior and Personality: an international journal*, vol. 47, no. 6, pp. 1–12, may 2019. doi: 10.2224/sbp.7567. [Online]. Available: <https://doi.org/10.2224/sbp.7567>
- [18] F. Niederman, M. Kaarst-Brown, J. Quesenberry, and T. Weitzel, "The future of it work: Computers and people," in *Proceedings of the 2019 on Computers and People Research Conference*, ser. SIGMIS-CPR '19. New York, NY, USA: Association for Computing Machinery, 2019. doi: 10.1145/3322385.3322403. ISBN 9781450360883 p. 28–34. [Online]. Available: <https://doi.org/10.1145/3322385.3322403>
- [19] B. Teehankee, "Humanistic entrepreneurship: An approach to virtue-based enterprise," *Asia-Pacific Social Science Review*, vol. 8, no. 1, pp. 89–110, 2008.
- [20] K. Larsen and D. Eargle. (2011) Organizational culture theory. [Online]. Available: https://is.theorizeit.org/wiki/Organizational_culture_theory
- [21] E. H. Schein, "Organizational culture," in *Working paper (Sloan School of Management)*, no. 2088-88. Sloan School of Management, Massachusetts Institute of Technology, 1988. [Online]. Available: <http://hdl.handle.net/1721.1/2224>
- [22] E. A. Martinez, N. Beaulieu, R. Gibbons, P. Pronovost, and T. Wang, "Organizational culture and performance," *American Economic Review*, vol. 105, no. 5, pp. 331–35, May 2015. doi: 10.1257/aer.p20151001. [Online]. Available: <https://www.aeaweb.org/articles?id=10.1257/aer.p20151001>
- [23] C. Packer, "A framework for the organizational assumptions underlying safety culture," International Atomic Energy Agency (IAEA), Tech. Rep., 2002. [Online]. Available: http://inis.iaea.org/search/search.aspx?orig_q=RN:34007162
- [24] M. Patnaik and B. Pattanaik, "Performance evaluation of employees in public sector banks," in *IEEE-International Conference On Advances In Engineering, Science And Management (ICAESM -2012)*. Institute of Electrical and Electronics Engineers, 2012, pp. 19–25.
- [25] R. J. Spang and N. D. Spang, "Human performance, error precursors and the tool kit," in *2020 IEEE IAS Electrical Safety Workshop (ESW)*, 2020. doi: 10.1109/ESW42757.2020.9188332 pp. 1–8.
- [26] I. Chiavenato, *Desempenho humano nas empresas*, 8th ed., ser. Série recursos humanos. Atlas, 2022. ISBN 9786559770458
- [27] S. J. Perkins, "Processing developments in employee performance and reward," *Journal of Organizational Effectiveness: People and Performance*, vol. 5, no. 3, pp. 289–300, 2018. doi: 10.1108/JOEPP-07-2018-0049. [Online]. Available: <https://doi.org/10.1108/JOEPP-07-2018-0049>
- [28] "Employee performance appraisals: Investigating the administrative, social and psychological nature of employee review," *Human Resource Management International Digest*, vol. 27, no. 5, pp. 38–40, 2019. doi: 10.1108/HRMID-05-2019-0130. [Online]. Available: <https://doi.org/10.1108/HRMID-05-2019-0130>
- [29] A. Tamayo and T. Paschoal, "A relação da motivação para o trabalho com as metas do trabalhador," *Revista de Administração Contemporânea*, vol. 7, no. 2, pp. 33–54, 2003. doi: 10.1590/S1415-6552003000400003
- [30] J. B. Miner, *Organizational behavior I. Essential theories of motivation and leadership*. M.E. Sharpe, Inc., 2005.
- [31] A. Shafagatova and A. V. Looy, "Developing a tool for process-oriented appraisals and rewards: Design science research," *Journal of Software: Evolution and Process*, vol. 33, no. 3, oct 2020. doi: 10.1002/smr.2321. [Online]. Available: <https://doi.org/10.1002/smr.2321>
- [32] A. Shafagatova and A. Van Looy, "A conceptual framework for process-oriented employee appraisals and rewards," *Knowledge and Process Management*, vol. 28, no. 1, pp. 90–104, 2021. doi: <https://doi.org/10.1002/kpm.1644>. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/kpm.1644>
- [33] Z. Yihua and W. Yuan, "Research and application of the data mining technology on the modern enterprise performance evaluation system," in *2009 International Conference on Information Management, Innovation Management and Industrial Engineering*, vol. 4. Institute of Electrical and Electronics Engineers, 2009. doi: 10.1109/ICIM.2009.470 pp. 34–39.
- [34] C. Huibao and L. Lei, "The study on appraisal of enterprise employee performance," in *2009 First International Workshop on Database Technology and Applications*. Institute of Electrical and Electronics Engineers, 2009. doi: 10.1109/DBTA.2009.45 pp. 632–637.
- [35] E. Miller, "The performance appraisal," *IEEE Potentials*, vol. 16, no. 2, pp. 20–21, 1997. doi: 10.1109/MP.1997.582455
- [36] A. S. De Oliveira Góes and R. C. L. De Oliveira, "A process for human resource performance evaluation using computational intelligence: An approach using a combination of rule-based classifiers and supervised learning algorithms," *IEEE Access*, vol. 8, pp. 39403–39419, 2020. doi: 10.1109/ACCESS.2020.2975485
- [37] E. Steelwater, "Humanism," in *Encyclopedia of Applied Ethics (Second Edition)*, 2nd ed., R. Chadwick, Ed. San Diego: Academic Press, 2012, pp. 674–682. ISBN 978-0-12-373932-2. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/B9780123739322002088>
- [38] F. Farah, "A Ética da avaliação de desempenho," Master's thesis, EAESP/FGV, São Paulo, 2000.
- [39] O. das Nações Unidas. (1948) Declaração universal dos direitos humanos. [Online]. Available: https://www.ohchr.org/en/udhr/documents/udhr_translations/por.pdf

- [40] C.-j. Wang, H.-m. Xu, and M.-h. Jiang, "Research on the dimensions and influencing factors of enterprise humanism management — an empirical study based on the questionnaire of dongguan enterprises," in *2020 16th International Conference on Computational Intelligence and Security (CIS)*. Institute of Electrical and Electronics Engineers, 2020. doi: 10.1109/CIS52066.2020.00044 pp. 169–173.
- [41] C. Dierksmeier, "What is 'humanistic' about humanistic management?" *Humanistic Management Journal*, vol. 1, no. 1, pp. 9–32, Sep 2016. doi: 10.1007/s41463-016-0002-6. [Online]. Available: <https://doi.org/10.1007/s41463-016-0002-6>
- [42] D. Melé, "The challenge of humanistic management," *Journal of Business Ethics*, vol. 44, no. 1, pp. 77–88, Apr 2003. doi: 10.1023/A:1023298710412. [Online]. Available: <https://doi.org/10.1023/A:1023298710412>
- [43] B. Teehanke. (2021) Principles and practices of humanistic management. [Online]. Available: <https://researchoutreach.org/articles/principles-and-practices-of-humanistic-management/>
- [44] F. Rapp, "Humanism and technology: The two-cultures debate," *Technology in Society*, vol. 7, no. 4, pp. 423–435, 1985. doi: [https://doi.org/10.1016/0160-791X\(85\)90009-0](https://doi.org/10.1016/0160-791X(85)90009-0). [Online]. Available: <https://www.sciencedirect.com/science/article/pii/0160791X85900090>
- [45] D. Messner, "Redefining and renewing humanism in the digital age [opinion]," *IEEE Technology and Society Magazine*, vol. 39, no. 2, pp. 35–40, 2020. doi: 10.1109/MTS.2020.2991498
- [46] M. Y. Vardi, "To serve humanity," *Commun. ACM*, vol. 62, no. 7, p. 7, jun 2019. doi: 10.1145/3338092. [Online]. Available: <https://doi.org/10.1145/3338092>
- [47] T. D. H. Initiative. (2019) Vienna manifesto on digital humanism. [Online]. Available: <https://dighum.ec.tuwien.ac.at/dighum-manifesto/>
- [48] K. M. Bartol and D. C. Martin, "Managing information systems personnel: A review of the literature and managerial implications," *MIS Quarterly*, vol. 6, pp. 49–70, 1982. [Online]. Available: <http://www.jstor.org/stable/248991>
- [49] B. Prommegger, D. Arshad, and H. Krcmar, "Understanding boundaryless it professionals: An investigation of personal characteristics, career mobility, and career success," in *Proceedings of the 2021 on Computers and People Research Conference*, ser. SIGMIS-CPR'21. New York, NY, USA: Association for Computing Machinery, 2021. doi: 10.1145/3458026.3462162. ISBN 9781450384063 p. 51–59. [Online]. Available: <https://doi.org/10.1145/3458026.3462162>
- [50] L. E. Potter, "Preparing for projects: It student self-evaluation of technical and professional skills," in *Proceedings of the 2020 on Computers and People Research Conference*, ser. SIGMIS-CPR'20. New York, NY, USA: Association for Computing Machinery, 2020. doi: 10.1145/3378539.3393868. ISBN 9781450371308 p. 63–69. [Online]. Available: <https://doi.org/10.1145/3378539.3393868>
- [51] B. Prommegger, M. Wiesche, and H. Krcmar, "What makes it professionals special? a literature review on context-specific theorizing in it workforce research," in *Proceedings of the 2020 on Computers and People Research Conference*, ser. SIGMIS-CPR'20. New York, NY, USA: Association for Computing Machinery, 2020. doi: 10.1145/3378539.3393861. ISBN 9781450371308 p. 81–90. [Online]. Available: <https://doi.org/10.1145/3378539.3393861>
- [52] M. Wiesche, D. Joseph, M. Ahuja, M. B. Watson-Manheim, and N. Langer, "The future of the it workforce," in *Proceedings of the 2019 on Computers and People Research Conference*, ser. SIGMIS-CPR '19. New York, NY, USA: Association for Computing Machinery, 2019. doi: 10.1145/3322385.3322409. ISBN 9781450360883 p. 12–13. [Online]. Available: <https://doi.org/10.1145/3322385.3322409>
- [53] F. Niederman, T. W. Ferratt, and E. M. Trauth, "On the co-evolution of information technology and information systems personnel," *SIGMIS Database*, vol. 47, no. 1, p. 29–50, feb 2016. doi: 10.1145/2894216.2894219. [Online]. Available: <https://doi.org/10.1145/2894216.2894219>
- [54] L. E. C. Potter, L. A. von Hellens, and S. H. Nielsen, "Childhood interest in it and the choice of it as a career: The experiences of a group of it professionals," in *Proceedings of the Special Interest Group on Management Information System's 47th Annual Conference on Computer Personnel Research*, ser. SIGMIS CPR '09. New York, NY, USA: Association for Computing Machinery, 2009. doi: 10.1145/1542130.1542138. ISBN 9781605584270 p. 33–40. [Online]. Available: <https://doi.org/10.1145/1542130.1542138>
- [55] M. W. Allen, D. J. Armstrong, M. F. Reid, and C. K. Riemenschneider, "It employee retention: Employee expectations and workplace environments," in *Proceedings of the Special Interest Group on Management Information System's 47th Annual Conference on Computer Personnel Research*, ser. SIGMIS CPR '09. New York, NY, USA: Association for Computing Machinery, 2009. doi: 10.1145/1542130.1542148. ISBN 9781605584270 p. 95–100. [Online]. Available: <https://doi.org/10.1145/1542130.1542148>
- [56] M. P. Zylka, "Putting the consequences of it turnover on the map: A review and call for research," in *Proceedings of the 2016 ACM SIGMIS Conference on Computers and People Research*, ser. SIGMIS-CPR '16. New York, NY, USA: Association for Computing Machinery, 2016. doi: 10.1145/2890602.2890618. ISBN 9781450342032 p. 87–95. [Online]. Available: <https://doi.org/10.1145/2890602.2890618>
- [57] F. Niederman and G. Crosetto, "Valuing the it workforce as intellectual capital," in *Proceedings of the 1999 ACM SIGCPR Conference on Computer Personnel Research*, ser. SIGCPR '99. New York, NY, USA: Association for Computing Machinery, 1999. doi: 10.1145/299513.299659. ISBN 1581130635 p. 174–181. [Online]. Available: <https://doi.org/10.1145/299513.299659>
- [58] A. Chizzotti, "HUMANISMO, EDUCAÇÃO e TECNOLOGIA," *Revista e-Curriculum*, vol. 18, no. 2, pp. 489–500, jun 2020. doi: 10.23925/1809-3876.2020v18i2p489-500. [Online]. Available: <https://doi.org/10.23925/1809-3876.2020v18i2p489-500>
- [59] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design science in information systems research," *MIS Quarterly*, vol. 28, no. 1, pp. 75–105, 2004. [Online]. Available: <http://www.jstor.org/stable/25148625>
- [60] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, "A design science research methodology for information systems research," *Journal of Management Information Systems*, vol. 24, no. 3, pp. 45–77, 2007. doi: 10.2753/MIS0742-1222240302. [Online]. Available: <https://doi.org/10.2753/MIS0742-1222240302>

4th Workshop on Data Science in Health, Ecology and Commerce

DATA Science in Health, Ecology and Commerce is a forum on all forms of data analysis, data economics, information systems and data based research, focusing on the interaction of those three fields. Here, data-driven solutions can be generated by understanding complex real-world (health) related problems, critical thinking and analytics to derive knowledge from (big) data. The past years have shown a forthcoming interest on innovative data technology and analytics solutions that link and utilize large amounts of data across individual digital ecosystems. First applications scenarios in the field of health, smart cities or agriculture merge data from various IoT devices, social media or application systems and demonstrate the great potential for gaining new insights, supporting decisions or providing smarter services. Together with inexpensive sensors and computing power we are ahead of a world that bases its decisions on data. However, we are only at the beginning of this journey and we need to further explore the required methods and technologies as well as the potential application fields and the impact on society and economy. This endeavor needs the knowledge of researchers from different fields applying diverse perspectives and using different methodological directions to find a way to grasp and fully understand the power and opportunities of data science.

This is a joint track by WIG2, the Scientific Institute for health economics and health service research, the Information Systems Institute of Leipzig University and the Helmholtz Center for Environmental Research.

TOPICS

We embrace a rich array of issues on data science and offer a platform for research from diverse methodological directions, including quantitative empirical research as well as qualitative contributions. We welcome research from a medical, technological, economic, political and societal perspective. The topics of interest therefore include but are not limited to:

- Data analysis in health, ecology and commerce
- Data analysis in climate change adaptation
- Data analysis in commerce
- (Health) Data management
- Health economics
- 5G(/6G) in health
- Data economics
- Data integration
- Semantic data analysis

- AI based data analysis
- Data based health service research
- Smart Service Engineering
- Integrating data in integrated care
- AI in integrated care
- Spatial health economics
- Risk adjustment and Predictive modelling
- Privacy in data science

TECHNICAL SESSION CHAIRS

- **Franczyk, Bogdan**, University of Leipzig, Germany
- **Militzer-Horstmann, Carsta**, WIG2 Institute for health economics and health service research, Leipzig, Germany
- **Häckl, Dennis**, University of Leipzig, Germany and WIG2 Institute for health economics and health service research, Leipzig, Germany
- **Bumberger, Jan**, Helmholtz-Centre for Environmental Research – UFZ, Germany
- **Reinhold, Olaf**, University of Leipzig / Social CRM Research Center, Germany

PROGRAM COMMITTEE

- **Alpkoçak, Adil**, Dokuz Eylul University
- **Ansari, Alireza**, Leipzig University, Germany and IORA Regional Center for Science and Technology Transfer, Iran
- **Arruda Filho, Emílio José Montero**, Universidade Federal do Para and University of Amazon, Brazil
- **Dey, Nilanjan**, Techno India College of Technology, India
- **Fass, Eric**, WIG2 Institute for Health Economics and Health Service Research
- **Hernes, Marcin**, Wrocław University of Economics and Business, Poland
- **Kozak, Karol**, Fraunhofer and Uniklinikum Dresden, Germany
- **Müller, Marco**, WIG2 Institute for Health Economics and Health Service Research
- **Popowski, Piotr**, Medical University of Gdańsk, Poland
- **Rot, Artur**, Wrocław University of Economics and Business, Poland
- **Sachdeva, Shelly**, National Institute of Technology Delhi, India
- **Siennicka, Agnieszka** Wrocław Medical University, Poland

- **Timpel, Patrick**, WIG2 Institute for Health Economics and Health Service Research, Leipzig, Germany and Technical University Dresden, Germany
- **Wasielewska-Michniewska, Katarzyna**, Systems Research Institute of the Polish Academy of Sciences, Poland

Encoder-Decoder Neural Network with Attention Mechanism for Types Detection in Linked Data

Oussama Hamel, Messaouda Fareh
 LRDSI Laboratory, Faculty of Sciences,
 University Blida 1,
 B.P 270, Route de Soumaa, BLIDA, ALGERIA
 Email: {oussamahamel09, farehm}@gmail.com

Abstract—With the emergence of use of Linked Data in different application domains, several problems have arisen, such as data incompleteness. Type detection for entities in RDF data is one of the most important tasks in dealing with the incompleteness of Linked Data. In this paper, we propose an approach based on Deep Learning techniques, using an encoder-decoder model with attention mechanism, embedding layer to extract the features of each subject from the RDF triples and the GRU cells to address the problem of vanishing. We use the DBpedia dataset for the training and test phases. Initial test results showed the effectiveness of our model.

I. INTRODUCTION

IN RECENT years, the Linked Open Data (LOD) cloud has been increasing in popularity. As a result of the success of the LOD, many semantic datasets are freely available on the Web in machine-understandable format (primarily RDF (Resource Description Framework)) related to different domains.

With the emergence of the semantic web and Linked Data, several problems related to data uncertainty have emerged, such as imprecision, incompleteness, etc. The main reason for the appearance of these problems lies in the way of building datasets. These were built from incomplete data, heterogeneous formats, semi-structured data, etc. The anomalies cited above pose problems when using so-called uncertain data in reasoning, decision-making, the generation of new knowledge, etc.

According to [1], few approaches use links among datasets, so they can't able to exploit the endless possibilities with the full knowledge of the Semantic Web.

The approaches that exploit data from only one dataset, they stay below what is possible with Linked Data. The reason of this limitations and difficulties of links discovery in Linked Data applications are:

- The datasets are produced, kept or managed by different organizations in different schemas, models, locations, systems and licenses [2]. There is not any “centralized control system,” therefore, each publisher decides how to produce, manage and publish a dataset based on its needs and choices;
- The development of several applications which are independent of schema.
- The same real-world entities or relationships are referred with different URIs and names and in different languages,

while languages have synonyms and homonyms that make harder that automatic links detection.

- The datasets usually contain complementary information, e.g., consider two datasets about the same domain each modeling a different aspect of the domain. The commonalities between these datasets can be very few and this does not aid automated linking and integration.
- The datasets can contain data that are erroneous, incomplete, out-of-date or conflicting.
- In addition, scalability challenges lie in developing solutions that could exploit the whole LOD as background knowledge by following links autonomously.

To improve the quality of RDF data, we choose to treat incompleteness, more specifically type incompleteness. Indeed, predicting missing types for dataset subjects will provide us with a more complete dataset.

Therefore, the results provided by applications using these datasets will become better. Our solution uses the predicates and objects belonging to the subject to predict its type. With the use of the encoder-decoder model, we will guarantee to extract the semantic relations between predicates and objects. This will improve the accuracy of subject type prediction. The attention mechanism was used to assign high weights to inputs with high importance. In this study, we will work on the DBpedia dataset, applying an approach based on deep learning.

Deep learning techniques have been recently used in many research axes to resolve different types of problems, Artificial intelligence systems use deep learning to solve computational tasks and complex problems quickly [3]. These techniques are very appropriate for dealing with large datasets. They have the ability to analyse and interpret Linked Data, that require efficient and effective tools. So, deep learning techniques are considered to be the most reliable solution that deal with the context of Linked Data, presented by RDF model.

In this paper, we have proposed an encoder-decoder network for multi-labeling. This network incorporates a attention mechanism to model the links between data. Our approach aims to predict missing types for RDF entities using data from their triples.

The remainder of this paper is organized as follows: in Section II, we define some Linked Data concepts. In section III of our paper, we explore the various related works that deal

with the type detection problem. Section IV shows in detail our proposed approach. Section V describes the experimental setup, and Section VI reports the results, followed by a discussion of the results in Section VII. Finally, Section VIII shows the set of perspectives as well as the conclusion of our work.

II. BACKGROUND AND CONTEXT

In this section, we introduce the main principles of Linked Data, we briefly recall some necessary background knowledge including principles of Linked Data, uncertainty, incompleteness, and links detection. We will also look at some areas where linked data can be used to demonstrate its utility.

A. Linked Data

“Linked Data refers to a method of publishing structured data, so that it can be interlinked and become more useful through semantic queries, founded on HTTP, RDF and URIs” [4].

Linked Data is a design principle that presents links between RDF-formatted data published on the web rather than links between documents. This enables machines to explore the web and find other data using the links concept [5].

The various objects in this version of the web are identified by URIs (Uniform Resource Identifier).

Tim Berners-Lee proposed four rules for designing Linked Data, which are explained below:

- Use of URIs to identify objects and concepts.
- Use of the http protocol to allow humans to access sites/data.
- Use of semantic web standards to provide information relevant to URIs.
- Introducing links to other data to give more options when exploring.

B. Uncertainty in Linked Data

Data uncertainty represents the degree of reliability, inaccuracy and imprecision of the data. According to the W3C, uncertainty is either aleatory or epistemic [6].

- Aleatory: characterized by lack of information, incompleteness information, etc. from the world.
- Epistemic: it describes the non-systematic nature of the data (variability, irreducibility) and the natural variability of a system.

C. Incompleteness in Linked Data

Due that there is an overwhelming quantity of heterogeneous data on the web. Integration of data silos provided by the Linked Open Data community can provide information to curate this data and boost the Semantic Web field to its true potential. Nevertheless, even the largest graphs, for example DBpedia, suffer from incompleteness [7]. Linked Data within the enterprise can be plagued with issues of data incompleteness, inconsistency and noise.

Incompleteness in the context of Linked Data can take the form of missing information, incomplete triples, missing links between different resources, etc.

D. Links detection

Published data must be linked to other existing datasets. However, creating links between datasets requires careful analysis, given that the amount of data published is constantly increasing, the links discovery process must be automatic. Consequently, to efficiently build the Web of data, there must be solutions capable of linking data between different datasets of web of data, to detect missing links between data.

The link detection task for Linked Data is seen as a solution for the datasets incompleteness. This task enables us to discover new triples and improves the quality of data delivered to applications and systems using datasets built on Linked Data concepts.

E. Applications of linked data

The Semantic Web and linked data can have a significant impact on a wide range of applications. Here are a few examples:

- Medical field: medical and personal patient data is commonly stored in multiple incompatible systems [8]. This representation may cause issues when integrating the data (incompleteness, errors, etc.), resulting in inaccurate or completely false diagnoses. The use of linked data to represent medical data is viewed as a solution for easing data integration. The incompleteness problem can be solved by using type detection in linked data.
- E-commerce: based on the semantic web, agents collect product data from multiple stores in order to provide the best deals to customers. They can also perform auctions, negotiations, and contract drafting automatically (or semi-automatically).
- Knowledge management in an organization: semantic web techniques are used to provide the possibility of representing knowledge in the form of concepts, allowing leaders to have answers to semantic queries.

III. RELATED WORKS

In this part, we will explore a number of related works that deal with type prediction in RDF datasets.

A statistical heuristic link based type prediction mechanism, has been proposed in [9], this work was evaluated on DBpedia.

In [10], the authors propose a supervised hierarchical SVM classification approach for DBpedia by exploiting the contents of Wikipedia articles.

In [11], the authors propose a multi-label classification algorithm based on word embedding such as Word2Vec, FastText and GloVe in order to capture the semantic aspect between entities and relations.

Another approach named Class Assignment Detector proposed by [12] to detect correct and incorrect classes assignments for entities in RDF data by analyzing class characteristics.

The authors in [13] solved the type prediction problem by using the Twitter profiles of RDF entities. The data extracted from these profiles were used as features in training data in order to facilitate the prediction task.

Another approach proposed in [14] uses word embedding and network embedding to predict the infobox types for Wikipedia articles. This information is useful for the type generation procedure for RDF entities.

In the work done in [15], the authors propose a binary classifier using structural data and based on machine learning techniques to predict the types of RDF entities.

The work carried out in [16] consists in proposing an approach which deals with types prediction by text classification. Two classifiers have been proposed to achieve this task.

Analysis: After studying the different approaches proposed in the literature for type detection in Linked Data, we can deduce that:

- The majority of works do not take into account the semantic relations between the different components of the triples.
- Use of different techniques for links detection in related works, we cite: supervised hierarchical SVM classification, statistical heuristic, machine learning techniques, text classification and word embedding.
 - The SVM algorithm is not appropriate for large datasets and for high number of features.
 - The extraction of features by the programmer is the major disadvantage of machine learning algorithms. As a result, the data quality suffers. On the other hand, this task is performed automatically by neural networks. Deep learning allows for the extraction of more and significant features and thus produces better results.
 - Using word embedding and statistical heuristics to predict types does not allow for the extraction of more significant features from the inputs and the possible semantic relations between triples. This has a negative impact on the results quality.
- The results obtained can be improved by proposing other solutions.
- Several works test their proposed methods on a subset of DBpedia data, but the tested part is not specified in the research works.
- The extraction of semantic relationships between different types (classes) and resources is not addressed in related works.
- The proposed methods do not assign weights to triples based on their importance during the type detection task.

In order to achieve this goal, we propose our approach based on deep learning in order to explore semantic relationships between the different components of RDF triples. We base our choice on deep learning models' ability to learn from large amounts of data. We added the attention mechanism to give a weighting to inputs according to their importance.

Therefore, the main contributions of this paper are:

- A proposition of embedding model, which exploit the semantic relations between RDF triples.

- A neural network based multi-label classification model for predicting type of resource RDF,
- Using the attention mechanism to improve the quality of model. By assigning weights to triples, semantic relationships between RDF resources and types can be extracted.
- Numerical representation (RDF2Vec) of resources and predicates in the DBpedia dataset.
- The type detection task is approached as a sequence-to-sequence problem, where the inputs and outputs are long sequences.

IV. MATERIALS AND METHOD

Before delving into our approach and giving more details, we first start by showing where the missing types problem lies in the ontology proposed by the World Wide Web Consortium (W3C)¹ organization, for modeling uncertain knowledge in the semantic web. This context is presented in Fig. 1. We use deep learning techniques for incompleteness processing.

Detecting links in Linked Data is a solution to identify classes of resources and therefore find new links between data and minimize incompleteness.

The automatic detection of missing types will improve data quality and provide reliable answers to queries launched by the various applications that use Linked Data.

Our solution allows predicting types for resources belonging to RDF datasets based on predicates and object values. It is a model built using deep learning techniques. Google Collaboratory² was used for the training phase using the DBpedia dataset.

A. Modeling problem

Our solution consists in treating the link detection problem as a multi-label classification problem.

Multi-label Classification is the task of assigning data points to a set of classes or categories which are not mutually exclusive, meaning that a point can belong simultaneously to different classes. In multi-label classification, the examples are associated with a set of labels Y from a set of disjoint labels L , $Y \subseteq L$.

The inputs of our model represent the predicates (P_1, P_2, \dots, P_n) and objects (O_1, O_2, \dots, O_n) belonging to a subject S . These inputs are used to predict the output which represents the types of the subject S .

B. Construction steps

The different steps of our model construction are mentioned in Fig. 2.

1) **DBpedia Dataset:** In order to train our model, we will use the DBpedia dataset³. The latter is built according to the Linked Data principles.

¹<https://www.w3.org/>

²<https://colab.research.google.com/>

³<http://gaia.infor.uva.es/hdt/dbpedia2016-10/dbpedia2016-10.hdt>

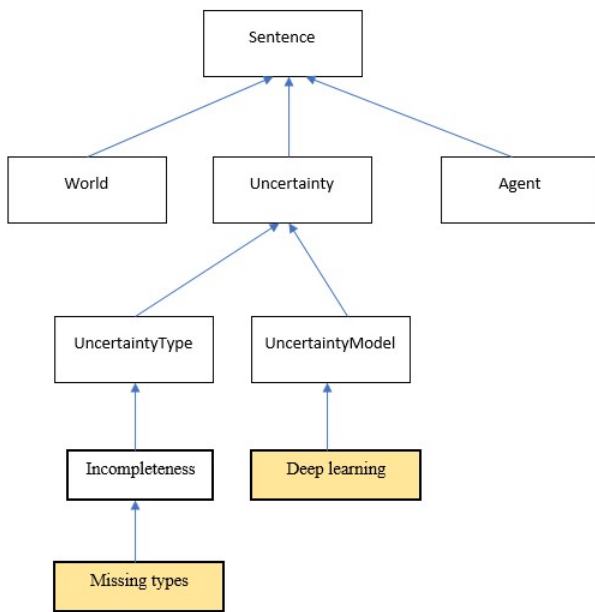


Fig. 1. Types detection in uncertainty ontology

Our dataset is composed of a set of triples (Subject, predicate, object), where the predicate represents the link between the subject and the object.

Our goal is to predict missing types (T_1, T_2, \dots, T_n) for subjects S_i based on their predicates (P_1, P_2, \dots, P_n) and objects (S_1, S_2, \dots, S_n) belonging to our dataset. i.e., for a given subject S_i , we use all its predicates and objects as inputs to predict its different classes or types. As a result, by inferring new triples, we can have a more complete dataset. (S_i, P_i, O_i) .

The first step of our approach is dataset reading, choosing the triples concerned by the different learning phases, as well as transforming the format of these triples into a numerical format.

Once our dataset is ready, we proceed to the pre-processing step, which consists of transforming the format of the triples into a format suitable for deep learning models (numerical representation).

Finally, we limit the number of predicates and objects belonging to each subject (inputs) to 205 objects, with the possible types as outputs.

2) **Pre-treatments:** Data preprocessing is an important or even crucial step for Machine Learning and Deep Learning. Data quality can directly affect the model learnability.

This step consists of transforming data into a more suitable format that can be used by the model. In this step we transformed each subject, object and predicate into numerical format. The numerical representation of inputs and outputs consists of giving numbers for each subject, predicate and

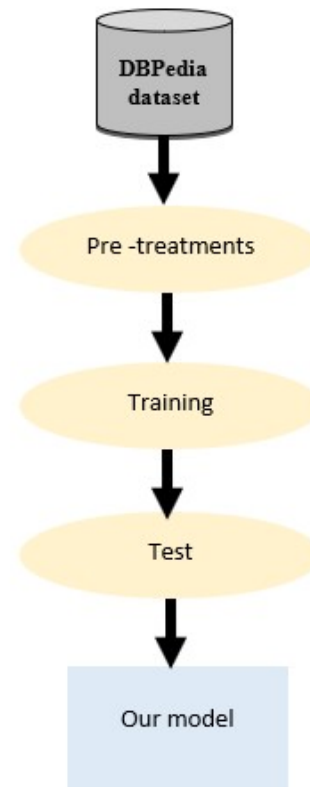


Fig. 2. Steps of our approach to type detection for Linked Data

object. Fig. 3 shows an example of this transformation.

C. Our model proposal

Our model is an encoder-decoder with attention mechanism which is a neural network design pattern that aids in the generation of an output sequence for every input sequence.

As shown in Fig. 4, the architecture is composed of two parts, encoder and decoder. Each part uses deep neural networks, more precisely gated recurrent units (GRU) in order to handle the sequence inputs of variable length.

The attention mechanism is a technique used in neural networks to focus on certain factors that can influence the



Fig. 3. An example of the preprocessing process

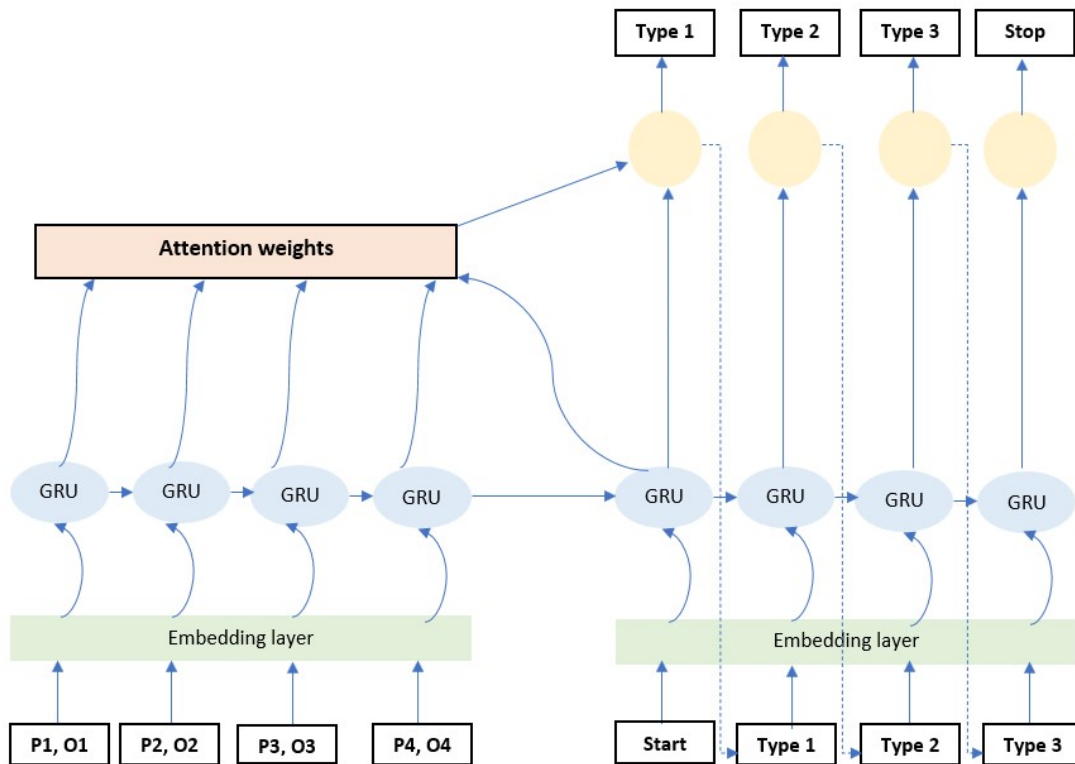


Fig. 4. Architecture of our encoder-decoder model with attention mechanism

model quality. The major contribution of this mechanism is to improve the sequence-to-sequence models performance. The details are mentioned in section IV-C4.

Our model uses as input the different predicates and objects belonging to the subject, and the types of the latter as output. We propose an embedding layer for each of the two components of our model (encoder and decoder).

1) **Encoder-Decoder:** The encoder-decoder architecture is used to generate output sequences from input sequences. In our model, we used GRU cells to build each component.

Our encoder is composed of two distinct modules: the Embedding layer, and the generated vector containing the information relating to inputs. This vector is used as the first hidden state of the decoder, in order to guide the decoder in its predictions. In what follows, we will detail how it works.

For our decoder, we distinguish two layers, a GRU layer and a SoftMax layer.

The GRU layer works the same way as the encoder layer with one exception based on Input/Output:

- The decoder takes as initial input the last hidden state generated by the encoder. This hidden state contains the essential information contained in each element of the input sentence.
- Just like the encoder, the decoder has an Embedding layer that generates the Embedding vectors from the numerical

representation of each subject.

- To generate a type T_i at a time step $T S_i$, the decoder takes as input: the hidden state, the output generated at the previous time step $T S_{i-1}$, as well as the Embedding vector.

The SoftMax layer predicts a probability distribution over the possible types, and choose the type with the highest probability.

2) **Embedding layer:** The embedding layer allow a reduced representation of inputs while keeping the semantic links between the subject's components. This layer enables the more features extraction from data.

3) **Gated recurrent units:** Gated recurrent neural networks GRU present a solution to the vanishing gradient problem. They have two gates, one for reset and another for update. They also use a hidden state mechanism, unlike LSTM which use a cell state and 3 gates. We will be able to process long sequences as a result of this.

4) **Attention mechanism:** The attention mechanism manages and quantifies the interdependence within input elements (Self-Attention) and between inputs and outputs (General Attention). This mechanism was introduced to solve one of the problems of sequence-to-sequence models, namely their

inability to provide good results when dealing with long sequences. This problem lies at the decoder level where only the last hidden state generated by the encoder is used as a context vector. An attention weight is generated for each input, giving high weights to the most important inputs in the type prediction phase. These values will be used by the decoder for the type prediction based on the results obtained from the SoftMax layer.

V. EXPERIMENTS

To evaluate the performance of our method, we use the DBpedia in our experimentation, to test the proposed method.

To this end, we use a subset of the dataset from the DBpedia.

A. Dataset

The DBpedia project is a community effort to extract structured information from Wikipedia and to make this information accessible on the Web.

DBpedia is a crowd-sourced community effort to extract structured, multilingual knowledge from the information created in various Wikimedia projects. The DBpedia knowledge bases are extracted from 125 Wikipedia editions. Altogether the DBpedia (2016-10) release consists of 13.1 billion pieces of information (RDF triples) out of which 1.7 billion were extracted from the English edition of Wikipedia, 6.6 billion were extracted from other language editions and 4.8 billion from Wikipedia Commons and Wikidata [17].

In this work, and due to technical constraints (RAM capacity), we used 15292 RDF triples for the different phases. These triples were divided as follows: 60% for the training phase, 20% for the validation phase and 20% for the test phase. the different details are mentioned in the Table I.

B. Hyper parameters

Our model uses the ADAM function as an optimizer and the 'Sparse Categorical Cross Entropy' cost function. The training phase consisted of 81 epochs. The various details are shown in Table II.

VI. RESULTS

To evaluate our method for type prediction in Linked Data, we use the standard evaluation measures: precision, recall and F-measure. In multi label classification, these criteria are defined in the following.

The metrics evaluate the multi-label classification system's performance, on each test example separately by comparing the predicted labels with the gold standard labels for each

TABLE I
THE NUMBER OF RECORDS FOR EACH STEP

Dataset	DBpedia
Training (60%)	9174
Validation (20%)	3059
Test (20%)	3059
Total (100%)	15292

TABLE II
HYPER PARAMETERS VALUES

Hyper parameter	Definition
Optimization function	Function ADAM
Loss function	Sparse Categorical Cross Entropy
Number of GRU Nodes	1024
Batch Size	64
Embedding size	256

test example. We focus on 3 major example-based metrics, as defined in [18] [19]:

$$Precision = \frac{1}{p} \sum_{i=1}^p \left(\frac{TP}{TP + FP} \right) \quad (1)$$

$$Recall = \frac{1}{p} \sum_{i=1}^p \left(\frac{TP}{TP + FN} \right) \quad (2)$$

$$F - measure = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \quad (3)$$

Where p is the number of instances in the test set. The true positives (TP) is defined as the labels that are identical to the gold standard labels, false positives (FP) as labels that are not true positives, and false negatives (FN) as the gold standard labels that were missed in the prediction results.

Table III illustrates the different results in two cases: Encoder-decoder model with Attention Mechanism (AM), witch represents our solution, and Encoder-decoder model without attention mechanism.

Histogram in Fig. 5 outline the evaluation results using the standard evaluation measures defined in Table III.

During the training phase, we obtained a cost function value of 0.0217 for our model with attention mechanism and 0.0937 for the model without attention mechanism.

After calculating the recall, precision, and F-measure values, we discovered that our model outperformed the encoder-decoder model without an attention mechanism.

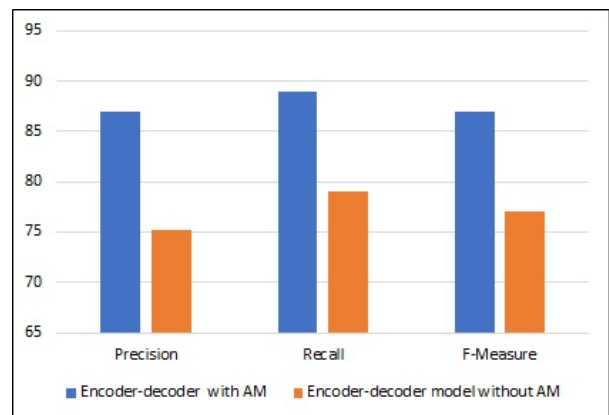


Fig. 5. Histogram of evaluation results

TABLE III
RESULTS OF TYPE PREDICTIONS WITH OUR MODEL AND A SIMPLE
ENCODER-DECODER MODEL

Model	Precision	Recall	F-Measure
Encoder-decoder model with AM	86.92	89.00	87.95
Encoder-decoder model without AM	75.16	79.02	77.04

VII. DISCUSSION

According to the presented results in Table III and Figure 5, we note that our modeling presents a good values for used evaluation metrics, comparing with encoder-decoder model without attention mechanism, witch is presented by 86.92% as precision value, 89.00% as Recall value and 87.95% as F-Measure value.

The results clearly demonstrated the significance of employing the attention mechanism. It enables the assignment of weights to inputs based on their importance. This increases the accuracy of output prediction. GRU cells positively influenced the results quality by making the features extraction task from inputs more efficient.

The results for the encoder-decoder model without an attention mechanism are less impressive because no attention value is assigned to the inputs to guide the type prediction process.

These results can be improved by running the model training phase on more powerful machines and by using larger datasets.

VIII. CONCLUSION

The semantic web community has been researching links detection and enrichment of Linked Data for last years. The volume of the available Linked Data on the web has been increasing considerably, along with the existence of erroneous, incomplete RDF data, kept this area active.

Links detection is a new research area of the Semantic Web which studies the problem of finding semantically related entities lying in different knowledge bases.

One of the most important challenges in Linked Data cloud is predicting the missing links between the entities, which is necessary to facilitate the inter-connectivity of datasets in the LOD cloud, in order to enhance and enrich the information that is known about them. Moreover, in the LOD cloud, information about the same entities is available in multiple datasets in different forms.

Links detection aims to deal with the issue of missing data. The completeness of the data simplifies decision-making and task performance in any field of application, including medical diagnostics, e-commerce and ecological prediction.

In this paper, we have proposed a promising new approach dealing with the type detection problem in Linked Data. We have treated this problem as a multi-label classification task using an encoder-decoder model with attention mechanism, and we have obtained very good results. However, in the

future, we want to validate our approach by testing it on different datasets and comparing it with the results of related works. We intend to propose a method for dealing with all possible semantic links. We also want to use NLP techniques on textual objects and to train our model on large datasets.

REFERENCES

- [1] P. Ristoski and H. Paulheim, "Semantic web in data mining and knowledge discovery: A comprehensive survey," *Journal of Web Semantics*, vol. 36, pp. 1–22, 2016.
- [2] M. Mountantonakis and Y. Tzitzikas, "Large-scale semantic integration of linked data: A survey," *ACM Computing Surveys (CSUR)*, vol. 52, no. 5, pp. 1–40, 2019.
- [3] R. A. Fiorini, "Computational intelligence from autonomous system to super-smart society and beyond," *International Journal of Software Science and Computational Intelligence (IJSSCI)*, vol. 12, no. 3, pp. 1–13, 2020.
- [4] C. Bizer, T. Heath, and T. Berners-Lee, "Linked data: The story so far," in *Semantic services, interoperability and web applications: emerging concepts*, pp. 205–227, IGI global, 2011.
- [5] T. Berners-Lee, "Linked data - design issues." <http://www.w3.org/DesignIssues/LinkedData.html>, 2006. Accessed: 2022-05-09.
- [6] K. J. Laskey and K. B. Laskey, "Uncertainty reasoning for the world wide web: Report on the urw3-xg incubator group.," *URSW*, vol. 8, pp. 108–116, 2008.
- [7] X. Sumba and J. Ortiz, "Between the interaction of graph neural networks and semantic web," in *Proceedings of the 2019 NeurIPS Workshop on Graph Representation Learning*, 2019.
- [8] C. Wilcox, S. Djahel, and V. Giagos, "Identifying the main causes of medical data incompleteness in the smart healthcare era," in *2021 International Symposium on Networks, Computers and Communications (ISNCC)*, pp. 1–6, IEEE, 2021.
- [9] H. Paulheim and C. Bizer, "Type inference on noisy rdf data," in *International semantic web conference*, pp. 510–525, Springer, 2013.
- [10] T. Kliegr and O. Zamazal, "Lhd 2.0: A text mining approach to typing entities in knowledge graphs," *Journal of Web Semantics*, vol. 39, pp. 47–61, 2016.
- [11] R. Biswas, R. Sofronova, M. Alam, and H. Sack, "Entity type prediction in knowledge graphs using embeddings," *arXiv preprint arXiv:2004.13702*, 2020.
- [12] M. Barati, Q. Bai, and Q. Liu, "An entropy-based class assignment detection approach for rdf data," in *Pacific rim international conference on artificial intelligence*, pp. 412–420, Springer, 2018.
- [13] Y. Nechaev, F. Corcoglioniti, and C. Giuliano, "Type prediction combining linked open data and social media," in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pp. 1033–1042, 2018.
- [14] R. Biswas, R. Türker, F. B. Moghaddam, M. Koutraki, and H. Sack, "Wikipedia infobox type prediction using embeddings.," in *DLAKGS@ESWC*, pp. 46–55, 2018.
- [15] N. Mihindukulasooriya and M. Rico, "Type prediction of rdf knowledge graphs using binary classifiers with structural data," in *International Conference on Web Engineering*, pp. 279–287, Springer, 2018.
- [16] X. Zhang, E. Lin, and S. Pi, "Predicting object types in linked data by text classification," in *2017 Fifth International Conference on Advanced Cloud and Big Data (CBD)*, pp. 391–396, IEEE, 2017.
- [17] H. Jin, C. Li, J. Zhang, L. Hou, J. Li, and P. Zhang, "Xlore2: large-scale cross-lingual knowledge graph construction and application," *Data Intelligence*, vol. 1, no. 1, pp. 77–98, 2019.
- [18] M.-L. Zhang and Z.-H. Zhou, "A review on multi-label learning algorithms," *IEEE transactions on knowledge and data engineering*, vol. 26, no. 8, pp. 1819–1837, 2013.
- [19] J. Du, Q. Chen, Y. Peng, Y. Xiang, C. Tao, and Z. Lu, "MI-net: multi-label classification of biomedical texts with deep neural networks," *Journal of the American Medical Informatics Association*, vol. 26, no. 11, pp. 1279–1285, 2019.

Model of Trust Dissemination of Products Based on Fuzzy Aggregation Norms

Oleksandr Sokolov

Faculty of Physics, Astronomy and Informatics,
 Nicolaus Copernicus University in Torun
 Grudziadzka 5, 87-100 Torun, Poland
 Email: osokolov@fizyka.umk.pl
 ORCID 0000-0002-6531-2203

Aleksandra Mrela

Institute of Informatics
 Kazimierz Wielki University in Bydgoszcz,
 Chodkiewicza 30, 85-064 Bydgoszcz, Poland
 Email: a.mrela@ukw.edu.pl
 ORCID 0000-0002-2059-864X

Maryla Bieniek-Majka

Institute of Law and Economics
 Kazimierz Wielki University in Bydgoszcz
 Chodkiewicza 30, 85-064 Bydgoszcz, Poland
 Email: maryla@ukw.edu.pl
 ORCID 0000-0003-1448-7406

Veslava Osinska

Department of Information Space Research
 Institute of Information & Communication Research
 Nicolaus Copernicus University in Torun
 ul. Wladyslawa Bojarskiego 1, Torun, Poland
 Email: wiewo@umk.edu.pl
 ORCID 0000-0002-1306-7832

Wlodzislaw Duch

Faculty of Physics, Astronomy and Informatics,
 Nicolaus Copernicus University in Torun
 Grudziadzka 5, 87-100 Torun, Poland
 Email: duch@fizyka.umk.pl
 ORCID 0000-0001-7882-4729

Abstract—Companies and even institutions have recently tried to form a reasonable opinion about their products. One way companies can achieve this is to prepare and distribute advertisements. Some advertising media, such as TV commercials, are costly and tiresome, so understanding and modeling some aspects of disseminating product information might be helpful. The article presents a multi-agent model of spreading trust in the product among agents based on fuzzy aggregation norms. When people with a similar opinion about a product discuss it with others or listen to a commercial, their trust increases, so the optimistic fuzzy aggregation norm is applied. When people with different opinions meet, their faith decreases, so the pessimistic norm is applied. In addition, the paper presents a theoretical example of this model application, namely, showing the results of a multi-agent model of spreading confidence in the product and presenting the results of the NetLogo simulation.

Index Terms—trust dissemination, multi-agent model, fuzzy aggregation norms, NetLogo application

I. INTRODUCTION

THE advertisement is present everywhere and slowly has been filling up people's everyday reality, and it is one of the systematic business processes where companies use technology. The methods used for helping people solve different business problems and based on information systems that start as practical business applications should also be analyzed at the conceptual level at the beginning by considering case studies, for example [1], [2].

This paper aims to present a theoretic model of spreading information about a product using fuzzy aggregation norms applicable in the area of product advertisement. The article also deals with searching for a method to observe the dissemination of knowledge about products applying the concept of an agent or an intelligent agent to analyze, model, and visualize the considered process using the NetLogo application. Multiagents' idea is used to analyze customer service, sales, and production scheduling [3], [4].

II. FUZZY LOGIC

When people describe the product, they may use words on the two extreme levels of the scale of product appreciation, like perfect or worst, or some intermediate terms like good, excellent, or wrong. Hence, fuzzy logic is much better to apply than classical logic [5].

L. Zadeh [6] introduced the concept of fuzzy sets. Let X be a non-empty space. Then $A \subseteq X$ is called a fuzzy set if it is a set of pairs $\{(x, \mu_A(x)), x \in X\}$, where $\mu_A : X \rightarrow [0, 1]$ is a membership function. Value $\mu_A(x)$ describes the level of membership of element x to set A .

III. THE FUNCTION OF TRUST DISSEMINATION

The article deals with the dissemination of products of knowledge or the reputation of scientists. Assume that we are

interested in the advertisement for one product, which will be called later the product.

Let \mathcal{A} denote the set of agents - people who might have some knowledge about the product and some level of trust about its quality. Let $T : \mathcal{A} \rightarrow [0, 1]$ be called the trust function. Assume that $A \in \mathcal{A}$ and let us consider five cases:

- $T(A) = 1$, then the trust of agent A is positive and perfect, there is no way to change their minds about the product;
- $T(A) \in (0.5, 1)$, then the agent's trust in the product is high;
- $T(A) = 0.5$, then agent A does not have any knowledge or is perfectly indifferent to the product;
- $T(A) \in (0, 0.5)$, then the trust is low;
- $T(A) = 0$, then agent A does not have any knowledge, or they dislike the product.

Now we will consider the situation when agents meet and discuss the product or get some new information about it from, for example, a TV commercial.

IV. OPTIMISTIC FUZZY AGGREGATION NORMS

The essential characteristic of optimistic fuzzy aggregation norms is that the examined property level is not less than before.

Let $I = [0, 1]$. Then $S_o : I \times I \rightarrow I$ is called an optimistic fuzzy aggregation norm if it fulfills the following conditions for each $x, y \in I$:

- (O1) $S_o(0, 0) = 0$
- (O2) $S_o(x, y) = S_o(y, x)$
- (O3) $S_o(x, y) \geq \max\{x, y\}$

Let us notice that it can be easily seen that

- (O4) $S_o(x, 0) \geq x$ for each $x \in I$
- (O5) $S_o(1, 1) = 1$

Condition (O1) shows that if an agent's trust in the product is 0 and they do not get any new information about it, their faith stays on zero levels (as before). Condition (O2), called commutativity, says that the order of getting pieces of information about the product does not matter. Thanks to condition (O3), we know that the product trust value is not less than the level after the meeting with another agent or watching the ad.

Additionally, condition (O4) indicates that if the level of trust in the product is positive and if the agent does not get any new pieces of information or advertisement, then the level of confidence is not reduced.

Let us choose as an optimistic aggregation norm the following function $S_o(x, y) = x + y - xy$ for each $x, y \in I$. In this case, $S_o(x, 0) = x$ for each $x \in I$.

Let us consider an example. Assume that two agents A_1 and A_2 , with the level of trust $t_1 = 0.6$ and $t_2 = 0.7$ respectively, meet and discuss the quality of the product. To calculate the trust of the agents we apply optimistic fuzzy aggregation norm: $T(A_1) = S_o(A_1, A_2) = t_1 + t_2 - t_1 \cdot t_2 = 0.6 + 0.7 - 0.6 \cdot 0.7 = 0.88$. By (O2), we know that $T(A_2) = S_o(A_2, A_1) = S_o(A_1, A_2) = 0.88$. Hence, the trust of both agents is equal and higher than before the meeting. Let us notice that the trust in the product of both agents in the product is strengthened.

V. PESSIMISTIC FUZZY AGGREGATION NORMS

The essential characteristic of pessimistic fuzzy aggregation norms is that the examined property level is not higher than before.

Let $I = [0, 1]$. Then $S_p : I \times I \rightarrow I$ is called a pessimistic fuzzy aggregation norm if it fulfills the following conditions for each $x, y \in I$:

- (P1) $S_p(1, 1) = 1$
- (P2) $S_p(x, y) = S_p(y, x)$
- (P3) $S_p(x, y) \leq \min\{x, y\}$

Let us notice that it can be easily seen that for each $x \in I$ we have

- (P4) $S_p(x, 1) \leq x$
- (P5) $S_p(0, 0) = 0$

Condition (P1) shows that if an agent believes in a product entirely and they get any new information about it, their belief stays on the same one (maximal) level (as before). Condition (P2), called commutativity, says that the order of getting pieces of information about the product does not matter. Condition (P3) shows that if the level of belief in a given product is not higher than the level of confidence in this product before and the new piece of faith the agent gets while speaking to other agents or the source of advertisement (TV, radio). Additionally, condition (P4) indicates that if the level of belief in a product is positive and if the agent gets the new perfect piece of information or advertisement, then the value of confidence is not increased.

Let us choose as a pessimistic aggregation norm the following function $S_p(x, y) = xy$ for each $x, y \in I$. Then we can easily see that condition (P4) we can write as follows $S_p(x, 1) = x$.

Let us consider an example. Assume that A_1 and A_2 are agents with the level of trust $t_1 = 0.2$ and $t_2 = 0.4$, respectively. After meeting and discussing the quality of the product, we can calculate the trust of the agents applying the pessimistic fuzzy aggregation norm: $T(A_1) = S_p(A_1, A_2) = t_1 \cdot t_2 = 0.2 \cdot 0.4 = 0.08$. By (P2), $T(A_1) = T(A_2)$. Of course, the trust in the product is lower for both agents.

Assume that agent A , with the trust t , does not meet another agent to discuss the product; their faith in the product after some time will be smaller and equal to $T(A) = S_p(t, t)$.

Let us consider another example. Let agent A , with the trust $t = 0.3$, does not discuss the qualities of the product and does not listen to the advertisement, then after some time: $T(A) = S_p(t, t) = 0.3 \cdot 0.3 = 0.09$. As we can quickly notice, their trust in the product lowered. When people do not listen to any information about the product or do not discuss it with other people, their confidence in the merchandise decreases. There is one exception, perfectly convinced people. They do not need any advertisement or talking to other people about the product, and their trust is still on level 1: $T(A) = 1 \cdot 1 = 1$.

VI. MULTI-AGENT MODEL OF THE TRUST-IN-THE PRODUCT DISSEMINATION

Let A be an agent with the level of trust in the product equals $t \in [0, 1]$ in some period. To design the model of the

trust-in-the-product dissemination, let us consider three cases.

- If for a given period, the agent does not meet other agents to discuss the product and does not watch any advertisement; then we can assume that their trust in the product suppresses with the forgetting coefficient F belonging to interval $[0, 1]$.
- If in a given period, the agent is exposed to advertising of the product or is in the shop selling the product, we can assume that their trust in the product increases with the strengthening coefficient $S \in [0, 1]$, which value depends on the level of aggressiveness of this ad.
- If agent A meets other agents A_1, A_2, \dots, A_n with levels of trust in the product equal to t_1, t_2, \dots, t_n , respectively, then A 's confidence in the product also increases. Hence the strengthening coefficient M can be calculated as a maximum of all agents' trust in this merchandise, $M = \max\{t, t_1, t_2, \dots, t_n\}$.

Hence, for the next period of time, agent A trust in the product is calculated in the following way:

$$T(A) = \begin{cases} S_p(t, F) & \text{if } A \text{ has no contact with the advert} \\ S_o(t, S) & \text{if } A \text{ is exposed to the advert} \\ S_o(t, M) & \text{if } A \text{ has meetings with other agents} \end{cases}$$

Let us consider the example. Assume that agent A with the level of the trust in the product equals $t = 0.5$ in January. This agent goes for a winter holiday and does not discuss the product's properties and is not exposed to the advertisement of it in February, so the forgetting coefficient is equal to $F = 0.9$. In March, this agent takes part in the meeting with two agents A_1 and A_2 with the levels of trust in the product equal to $t_1 = 0.3$ and $t_2 = 0.7$ in March. Finally, this agent watches a fascinating film on TV with the product's advert in April, so the strengthening coefficient is equal to $S = 0.8$.

Let $t_{month} = T(A, \text{month})$ denotes this agent's level of trust in the product in the considered month. Hence, $t_{Jan} = T(A, \text{January}) = 0.5$. Next, $t_{Feb} = T(A, \text{February}) = S_p(t_{Jan}, F) = S_p(0.5, 0.9) = 0.5 \cdot 0.9 = 0.45$. Afterwards, $M = \max\{0.45, 0.3, 0.7\} = 0.7$ and $t_{March} = T(A, \text{March}) = S_o(t_{Feb}, M) = S_o(0.45, 0.7) = 0.45 + 0.7 - 0.45 \cdot 0.7 = 0.84$. Finally, $t_{Apr} = T(A, \text{April}) = S_o(t_{March}, S) = S_o(0.84, 0.8) = 0.84 + 0.8 - 0.84 \cdot 0.8 = 0.97$. Thus, the presented multi-agent model can simulate the levels of agents' trust in the product and the influence of other agents and the product advertising on the agents' trust level.

VII. SIMULATION OF THE MODEL OF THE TRUST DISSEMINATION

A multi-agent simulation of the trust dissemination of the product was developed in NetLogo's programmable modeling environment, in which applications coded in NetLogo are very convenient to observe agents (turtles) meeting and changing their opinions about the product.

The simulation started with the group of people represented by gray turtles showing no knowledge about the product and advertising media represented by green squares. The simulation follows the Ant Colony Optimization [7] algorithm, which

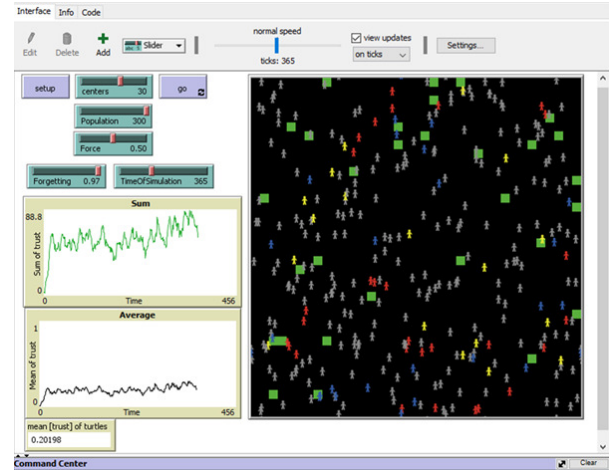


Fig. 1. The screenshot of the middle phase of the application developed for the simulation of the trust in the product

applies probability to solve some computational problems that require looking for paths while traversing plots.

During the simulation, people move, meet other people (during these meetings, they exchange opinions about the product), and meet advertising media (get some information about the product). Initially, the agents' trust in the product is 0, and the media's is 0.9. During the meetings, the product's trust changes according to the fuzzy aggregation norms. Hence, after some time, agents change their colors according to the following rules: they stay gray (the trust is smaller than 0.3), change color to yellow (the trust is between 0.3 and 0.6), - to blue (the trust is between 0.6 and 0.9) or - to red (the trust is greater than 0.9).

VIII. DISCUSSION

According to K. Kubiak, referring to A. Lubecka, the subject of advertising, has become one of the most frequently discussed issues [8]. Institutions that use advertising to influence the expected behavior use various methods of reaching a specific audience [9]. In advertising design, the most commonly used methods are: evoking intense emotional states, making promises (positive and negative), and using the image of a public figure. According to the theory of stereotypes, ad recipients remember these performances as relatively permanent. Time constraints or the size of the advertising space, combined with the willingness to communicate, impose the need to schematize (stereotype) the presented events, so stereotypes are used because they function on the scale of society [8].

Even the most straightforward messages affect the knowledge, emotions, or behavior of the recipient, sometimes regardless of the will of the message sender. However, there are cases of manipulating the other person. However, honesty towards the customer is not the primary determinant of the product promotion process for some advertisers. Persuasive and argumentative techniques may lead to stereotypes' consolidation in the social consciousness [10].

Advertisements influence emotions and affect product ratings and consumer attitudes. Scientists dealing with this topic pointed to the vital role of activating memories by advertising. The strength of emotional reactions also depends on the purpose of viewing the ad. In the minds of consumers, the goals of observing advertisements may be different; for example, some people want to get information about a product or have the pleasure of viewing the aesthetic images. Thanks to cognitive engagement, product information is processed more accurately, and emotional engagement results in a more positive attitude towards advertising. Therefore, referring to memories will increase involvement in processing information about an advertisement, influencing its evaluation and remembering [5].

In increasing skepticism about the content and declining credibility of messages about products and services, customers' opinions are more important. In recent years, word-of-mouth marketing, also known as gossip or recommendation marketing, is becoming more critical. Its popularity results from people's psychological construction because they need to share their opinions with others [14]. Therefore, the companies found their hope in word-of-mouth marketing, that is, a selfless recommendation of products, brands, or services among their families, friends, or strangers [13].

Word-of-mouth marketing is the oldest, best and cheapest marketing tool. When people tell the truth about a product, the effect of such advertising can be striking, but impersonating an ordinary user praising the product can lead to a loss of trust and turn against the trader [13]. Word of mouth marketing distinguishes into two forms, i.e., face-to-face marketing and online communication [11]. eWOM (electronic word of mouth marketing) has recently attracted much interest from researchers. However, with consumers' increasing dependence on information retrieval and the continued growth of social media, the importance of eWOM should not be overstated [12]. Regardless of the proportion of the phenomenon, the strength of adverse opinions is higher than positive ones. Therefore, the company's task is to strengthen positive thoughts, and weaken negative [14].

Being recognized is very important also for researchers. Scientists build their scientific profiles in only one or a few disciplines or scientific fields, and scientometricians try to find factors to estimate their contribution to a specific discipline or a field. One of the models of computer systems for calculating the contribution to research areas in computer science is presented in [15]. Moreover, bibliometricians try to visualize the scientist's contribution to disciplines based on published articles and journals' profiles by choosing the weights of examined fields empirically [16].

IX. CONCLUSIONS

We present a model based on fuzzy aggregation norms to describe spreading information about a product that can be applied in product advertisements. We also use several simulations to test the basic assumptions of the model. Selected

actions with predefined emotional states and parameters were simulated in the NetLogo environment.

Summing up, applying optimistic and pessimistic fuzzy norms has the following properties.

- 1) The situation with increasing and decreasing trust of the product among customers can be modeled and analyzed.
- 2) The influence of Advertising Centers can be modeled.
- 3) The role of word-of-mouth marketing and its influence on product trust dissemination can be described.
- 4) Fuzzy aggregation norms can be used to model the product trust dissemination in marketing.

REFERENCES

- [1] A. Martinez, B. Vazquez, H. Estrada, L. Santillan and Zavalaet, "Incorporating technology in service-oriented i* business models: a case study", *Information Systems and e-Business Management*, 15, 2017, pp. 461, <https://doi.org/10.1007/s10257-016-0316-9>
- [2] K. Borodako, J. Berbeka and M. Rudnicki, "Technology used in knowledge management by global professional event services", *JGIM*, vol. 29, no. 1, 2021, pp. 145-163. <http://doi.org/10.4018/JGIM.2021010108>
- [3] Y. S. Lee Y. S and R. Sikora, "Application of adaptive strategy for supply chain agent", *Information Systems and e-Business Management*, 17, 2019, p. 117, <https://doi.org/10.1007/s10257-018-0378-y>
- [4] J. Jin, C. Song, J. Li, K. Gai, J. Wang and W. Zhang, "Real-time bidding with multi-agent reinforcement learning in display advertising", *CIKM '18: Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, October 2018, pp. 2193–2201, <https://doi.org/10.1145/3269206.3272021>
- [5] A. Falkowski and A. Grochowska, "Influence of emotions on the evaluation and memory of advertising: Research in the paradigm of retrospective memory shaping", *Roczniki Psychologiczne*, vol. XI, 2, 2008, pp. 107–136, [in Polish].
- [6] L. Zadeh, "Fuzzy sets", *Information and Control*, vol. 8, 1965, pp. 338–353, [https://doi.org/10.1016/S0019-9958\(65\)90241-X](https://doi.org/10.1016/S0019-9958(65)90241-X)
- [7] M. Dorigo, "Ant colony optimization", T. Stutzle, Cambridge, Mass., MIT Press, ISBN 0-262-04219-3, OCLC 57182707, 2004.
- [8] K. Kubiak, "Selected socio-cultural concepts as a tool for analyzing advertising messages", in: K. Kubiak (ed.), "Social engineering advertising", Wyzsza Szkola Promocji, Mediow i Show Businessu, Warszawa, 2016, pp. 38–60, [in Polish].
- [9] J. Petrykowska, "Ways of influencing the recipients of social advertising", *Acta Universitatis Nicolai Copernici, Zarzadzanie XL - Zeszyt 413*, Torun, 2013, pp. 39–46, [in Polish].
- [10] E. Sanecka, "Psychological mechanisms of the impact of advertising and manipulation in advertising", 2010, <http://www publikacje.edu.pl/pdf/10023.pdf>, [in Polish]
- [11] B. Pilarczyk, "Innovations in marketing communication", *Zeszyty Naukowe PTE*, 9, Kraków, 2011, pp. 271–286 [in Polish].
- [12] R. A. King, P. Racherla and V. D. Bush, "What we know and what don't know about online word of mouth: A review and synthesis of the literature", *Journal of Interactive Marketing*, vol. 28, Issue 3, August 2014, pp. 167–183, <https://doi.org/10.1016/j.intmar.2014.02.001>
- [13] I. Osmolska, "Word-of-mouth or shady marketing", *Zeszyty Naukowe Wyzszej Szkoły Administracji i Biznesu im. E. Kwiatkowskiego w Gdyni*, Zeszyt 22, 2016, Zarzadzanie 1, Gdynia, pp. 301–315.
- [14] M. Gawronska M. (2013) 'Word-of-mouth marketing as a modern form of promotion', *Rynek-Spotecznostwo-Kultura*, 2 (6), 2013, pp. 30–35, [in Polish].
- [15] O. Sokolov, W. Osinska, A. Mrela and W. Duch, "Modeling of scientific publications disciplinary collocation based on optimistic fuzzy aggregation norm", in: J. Swiatek, L. Borzowski, Z. Wilimowska, (eds) *Information Systems Architecture and Technology: Proc. of 39th International Conference on Information Systems Architecture and Technology ISAT 2018*. ISAT 2018. *Advances in Intelligent Systems and Computing*, vol 853. Springer, Cham. <https://doi.org/10.1007/978-3-319-99996-8-14>
- [16] V. Osinska and P. Bala, "New methods for visualization and improvement of classification schemes: the case of computer science", *Knowledge Organization*, vol. 37, no 3, 2010, pp. 157–172, <https://doi.org/10.5771/0943-7444-2010-3-157>

Detecting Symptoms of Dementia in Elderly Persons using Features of Pupil Light Reflex

Minoru Nakayama*

* School of Engineering,
Tokyo Institute of Technology,
Tokyo, Japan 152-8552
Email: nakayama@ict.e.titech.ac.jp

Wioletta Nowak and Anna Zarowska[†]

[†] Biomedical Eng. and Instrumentation
Wrocław University of Science and Technology,
Wrocław, Poland 50–370
Email: wioletta.nowak, anna.zarowska@pwr.edu.pl

Abstract—A procedure for detecting cognitive impairment in senior citizens is examined using pupil light reflex (PLR) for chromatic light pulse and a portable measuring system. Features of PLRs of blue and red light pulses are compared. PLRs of elderly subjects were studied in order to develop a procedure for detection of the symptoms of cognitive function impairment using a dementia evaluation test. PLRs of both eyes were measured using blue and red light pulses aimed at either of the two eyes. The features of PLR waveforms for each eye were remained in comparable level for every group of participant. Three factor scores were calculated from the features, and a classification procedure for determining the level of dementia in a subject was created using regression analysis. As a result, the contribution of factor scores for blue light pulses according to a participant's age was confirmed.

Index Terms—Pupil, Pupil Light Reflex, Alzheimer's disease, feature extraction, logistic regression

I. INTRODUCTION

SYMPTOMS of cognitive function impairment are used to diagnose Alzheimer's Disease (AD) and mild cognitive impairment (MCI). A major diagnostic procedure is the Mini-Mental State Examination (MMSE), which is based on a set of face-to-face clinical tests. These require participants to have sufficient communication skills, however. Therefore, a quicker and easier objective procedures should be developed.

The study of conventional pupil light reflex (PLR) activity [1], [2] suggests that as this activity represents to visual information processing of retinal stimuli and the ability to activate neural signal transfers, it should be evaluated as an alternative means of diagnosing cognitive function impairment [3], [4]. Also, PLR responses based on Melanopsin ganglion cells [5], [6], [7] can be applied to the study of aged macular disease (AMD) and AD [5], [6], [8], and the possibility of their use in diagnosing these diseases has been studied [9], [10], [11], [12]. A simple procedure to detect AMD and AD patients is required for medical and clinical staff who treat elderly people [13]. In a sense, a diagnostic procedure using ocular-motors may be an easy way, as it does not require verbal communication.

This research was partially supported by the Japan Science and Technology Agency (JST), Adaptable and Seamless Technology transfer program through target driven R&D (A-STEP) [JPMJTM20CQ, 2020-2022].

The authors have been conducting feasibility studies about conducting PLR observations using a portable measuring system at clinical institutions. During the current survey, additional elderly people were invited to participate and their responses were analyzed. Estimation performance and validity were evaluated. In this paper, the following points are addressed.

- 1) Features of PLRs for blue and red light pulses of the left and right eyes are compared, and the differences are extracted.
- 2) The ability of classifying participants as AD/MCI or normal control (NC) using MMSE score and PLR features.
- 3) The contribution of the participant's age is also examined.
- 4) Prediction performances of participants with AD or MCI procedures are developed and evaluated.

II. METHOD

Pupil light reflex was observed in senior citizens who may be AD patients, pseudo-positive participants, or have no cognitive impairment i.e., normal.

A. Stimuli

Participants were introduced to a temporary dark space, where the 5 following experimental sessions were conducted for 10 seconds each.

- 1) Condition1: Control session without light pulses
- 2) Condition2: Blue light pulse to the right eye
- 3) Condition3: Blue light pulse to the left eye
- 4) Condition4: Red light pulse to the right eye
- 5) Condition5: Red light pulse to the left eye

The experiment is designed to study the influence of light pulses on synaptic connections between both eyes in response to light pulses to either eye. Light pulses transfer from retinal ganglion cells on the irradiated eye to sphincters of both eyes via the Edinger-Westphal Nucleus [14]. The processes of miosis and restoration were observed in all 4 session. A short break to be relax was inserted between each session.

B. Procedure

The size in pixels of pupil responses were measured at 60Hz using an equipment with blue and red light source as shown



Fig. 1. Equipment to observe pupillary changes

TABLE I
FEATURES OF PLR

Variables	Definitions
RA	Relative Amplitude of miosis
t_min	Time at minimum size
diff_min	Minimum differential of size
t_diff_min	Time at minimum differential
diff_max	Maximum differential of size
t_diff_max	Time at maximum differential
diff2_min	Minimum acceleration
t_diff2_min	Time at minimum acceleration
diff2_max	Maximum acceleration
t_diff2_max	Time at maximum acceleration

in Figure 1 (URATANI, HITOMIRU). The light sources were blue (469nm, 14.3cd/m², 6.5lx) and red (625nm, 12.3cd/m², 10.5lx). Both pupil sizes were measured over all conditions. Blink artifacts were removed manually after the measurement.

The experiment was conducted by a clinical physician at a medical institution, and the procedure was approved by an ethics committee at Osaka Kawasaki Rehabilitation University.

C. Participants

The valid data was obtained from 101 participants, 66 females and 35 males. Their mean age was 78.5 and the SD (standard deviation) was 8.9 years. Participants were selected at a medical institute and MMSE test was conducted. The results were classified into three groups according to MMSE scores. These were AD (Alzheimer's disease, with MMSE≤23), MCI(Mild cognitive impairment, with MMSE≤27) and others, whose conditions was NC(Normal Control). The distribution was as follows:

- AD: 31(F:21, M:10), Mean age:83.0, SD:6.3 years.
- MCI: 9(F:5, M:4), Mean age:82.1, SD:6.3 years.
- NC: 61(F:40, M:21), Mean age:75.6, SD:9.2 years.

As the age of participants may influence their condition, four age levels were created: less than 66 years old (0), 66-75 years old (1), 76-85 years old (2), higher than 85 years old (3). Though participants were older persons who might have some health problems, these points were not considered in the following analysis.

III. RESULTS

A. PLR waveforms

An example of PLR waveforms for a NC participant is shown in Figure 2. The horizontal axis represents time, and the

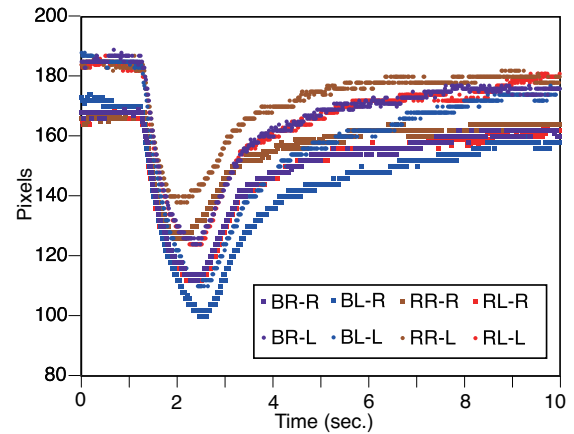


Fig. 2. Examples of PLR of both eyes for four conditions (NC participant, 76yo, M), categories:[light color][irradiated eye]-[observed eye]

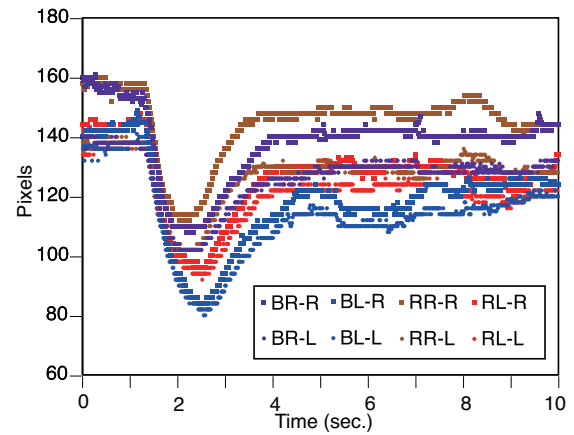


Fig. 3. Examples of PLR of both eyes for four conditions (AD participant, 87yo, F), categories:[light color][irradiated eye]-[observed eye]

vertical axis represents pupil size in pixels for the experimental conditions 2~5. There are some differences in pupil size between the left and right eyes at the initial point. The legend "BR-R" means that Right pupil response when Blue light irradiates to Right eye, and also "RL-R" means that Right pupil response when Red light irradiates to Left eye. Also, levels of contraction are different between conditions as in the previous work, which presented PLR responses to blue or red light pulses [6]. Another example of an AD patient is shown in Figure 3. Some typical features are observed such as deviation during the restoration process after the constriction. Several features of waveforms were extracted in order to compare groups of participants in relation to the previous study [12] as shown in Table I.

The first hypothesis is that there is a feature difference between the left and right eyes when light pulses are directed at either eye. The hypothesis was examined using a t-test of features of both eyes, such as between irradiated eye and non-irradiated eye. The features were extracted from standardized waveforms in order to reduce the potential differences.

TABLE II
FACTOR LOADING MATRIX FOR PLR FEATURES

Variables	Factor1	Factor2	Factor3
diff_min	0.87	-.13	0.09
diff2_min	0.76	0.06	0.16
diff2_max	-0.83	-.17	0.22
diff_max	-.36	0.08	0.15
RA	-.24	0.78	-.09
t_min	0.22	0.73	0.49
t_diff2_min	-.13	-.00	0.49
t_diff_min	-.05	-.03	0.36
t_diff_max	-.11	0.23	0.36
t_diff2_max	0.06	0.07	0.30

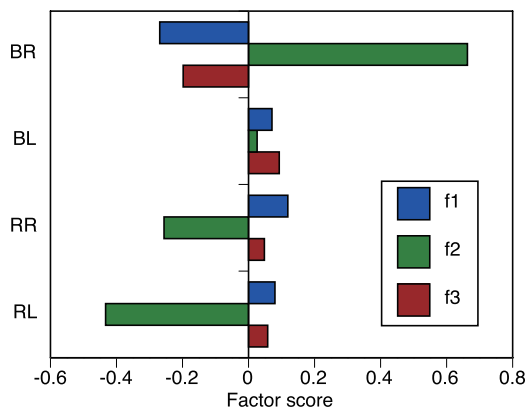


Fig. 4. Comparison of factor scores between stimuli (f1~f3: factor scores for Factor 1~3)

In the results of the test, there are no significant differences in any of the features. There were a few exceptions, but the results did not coincide with the results of either colour of light pulse.

B. Factor analysis and factor scores

Since every feature includes measurement errors and individual differences, the latent factors are extracted using factor analysis according to the method used in the previous study [12].

The results of factor analysis are shown as a factor loading matrix in Table II. In this paper, three factors are employed, and the overall contribution ratio of the factors is 45.5%. Factor 1 represents the differential rate and acceleration of pupillary change, Factor 2 represents the features of contraction such as relative amplitude and its time, and Factor 3 represents the times for the differential rates and acceleration, as mentioned in Factor 1.

Three factor scores are calculated using the factor loading matrix. When these scores of both eyes are compared, there are also no significant differences.

The factor scores for experimental conditions are summarized and compared in Figure 4. Changes in Factor-2 scores suggest a continuous decrease according to the experimental conditions. Also, there are significant differences in the three factor scores of blue and red stimuli. Within a colour stimulus

condition, there are significant differences in the three factor scores for blue light pulses, and significant differences in Factor-2 scores for red light pulses ($t(402) = 2.13, p < 0.05$). The differences between sessions using the same colour condition should be considered, in particular the differences for blue light should be evaluated separately.

In addition, the factor of age level on factor scores is examined using two-way ANOVA of participant groups and age levels. Though the factor for the participant groups is not significant, the factor for age level is significant for Factor-1 scores ($F(3, 801) = 19.9, p < 0.01$), and the interaction between the two factors (age and participant group) is also significant ($F(2, 801) = 9.4, p < 0.01$). Therefore, the factor of age may affect the differential and the acceleration of pupillary change as presented in Factor-1.

The influence of a subject's level of dementia on PLR features was not confirmed in the above analysis. The factors of stimulus light wavelength and the age of patient were significant and are the major components of the deviation. In order to examine the effectiveness of the extracted features for a prediction of cognitive function impairment, an estimation procedure using a logistic analysis which had been introduced in a previous study [12] was conducted. Here, both MCI and AD patients are merged as the "AD+MCI" group since the number of MCI participants is limited. The probability of cognitive impairment is calculated using factor scores for each participant. As there are no significant differences between eyes, averaged features of responses of both eyes are employed. In considering the differences between session stimuli, two sets of features for blue light conditions and averaged features for red light conditions are introduced, for a total of 9 variables altogether.

Table III shows a summary of several prediction models and AUC (Area under the Curve) as an index of accuracy of binary classification for a ROC (Receiver Operating Characteristic Curve). Since a threshold for the classification may depend on the diagnostic policy such as reducing False positive rate, the accuracy is evaluated using AUC. An example of ROC for Model 2 is illustrated in Figure 5. Model 1 consists of 9 factor scores, Models 2 and 3 include age level or age. Model 4 employs significant contributing variables using a stepwise selection technique. Participant's age information aids classification performance.

Probabilities for the classification of cognitive impairment based on Model-2 are calculated, and the relationships between MMSE scores and the probabilities are summarized in Figure 6. The horizontal axis represents MMSE scores, and the

TABLE III
PREDICTION MODELS USING FACTOR SCORES OF PLRS

Model	Variables	AUC
1	9 factor scores	0.77
2	9 factor scores + age group	0.84
3	9 factor scores + age	0.84
4	Selected variables: 5 factor scores + age group	0.84

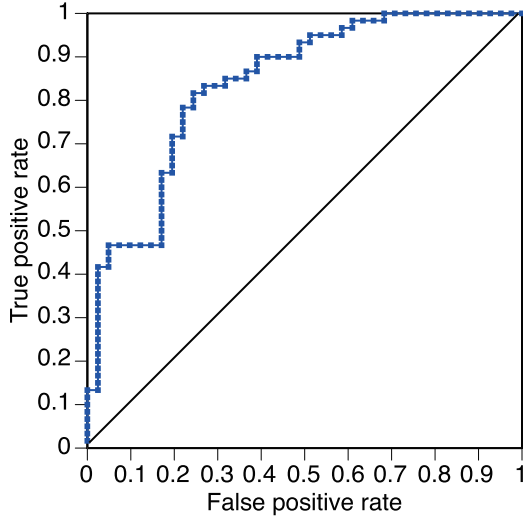


Fig. 5. ROC for Model 2 (AUC=0.84)

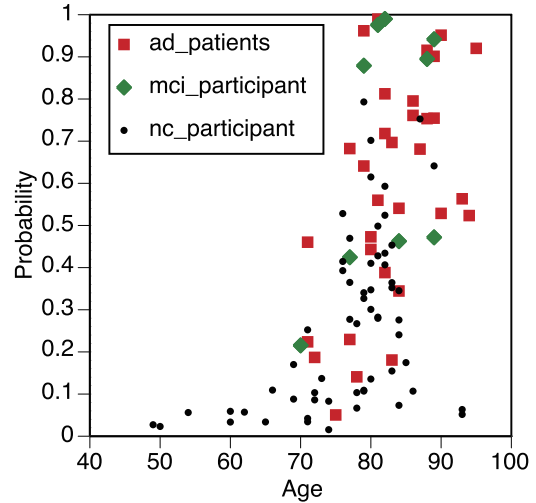


Fig. 7. Change in probabilities according to participant's age

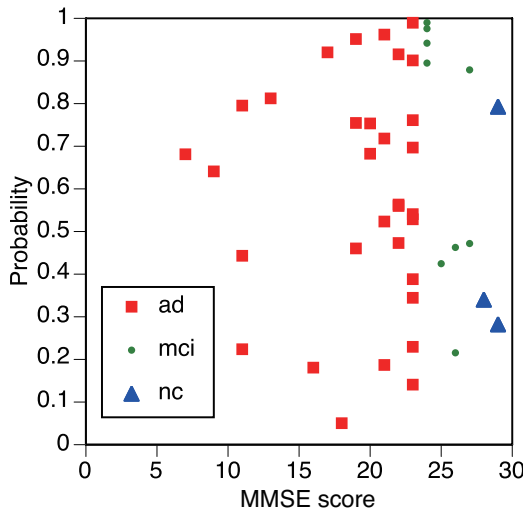


Fig. 6. Relationship between MMSE scores and computed probabilities

vertical axis represents the probability. Participants who were tested using MMSE are plotted in the figure according to their participant group, AD, MCI or NC. Confirmation of the contribution of a participant's age is shown in Figure 7, where the horizontal axis represents the age. In this figure, cognitive impairment can be observed in subjects over 70 years old, and the probability increases markedly from around 70 onwards. When the threshold for AD+MCI is set to 0.5, 80 percent of participants are classified correctly. The AUC is 0.84 as shown in Table III.

Mean probabilities for the groups of AD+MCI and NC by age level are summarized in Figure 8, in order to evaluate the contribution of age level. Overall, mean probabilities increase with age level. In particular, the probability of AD+MCI increases over age 75 while mean probability of NC remains under 0.5.

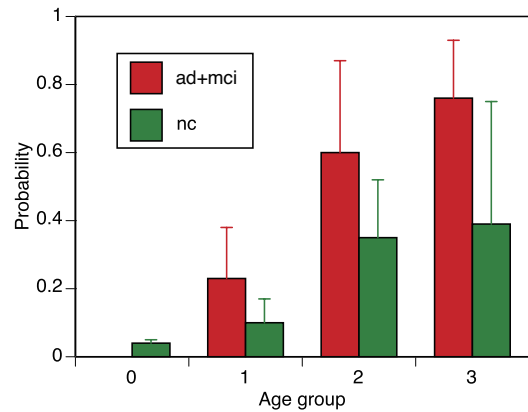


Fig. 8. Comparison of mean probabilities between AD+MCI and NC groups

C. Variable selection of the regression function

Model-4 in Table III was generated using a variable selection procedure. All 5 selected variables are factor scores for blue light pulses during two test sessions, and factor scores for red light pulses were not used. The fitting index AUC is comparable with the values of the other functions. This suggests the possibility that prediction can be made using responses to blue light pulses. More detailed points regarding this will be summarized in the following discussion section.

IV. DISCUSSION

In the first hypothesis, cognitive impairment may be affected by the influence of the oculomotor nerve, which connects retinal ganglion cells to the pretectal area on the synaptic path. However, no significant differences in the extracted features were observed during several chromatic light pulses, though there were some differences between the experimental sessions, as shown in Figure 2. One of the possible reasons is the dependence on the accuracy of measurement, because feature extraction is based on point estimation. In particular, the

small pupils of senior citizens may influence the measurement of temporal change of pupil size. Another problem may be that the extracted features focused on the constriction phase of PLRs without allowing for the restoration phase which follows. The PLR difference between the left and right eyes should be measured carefully in considering the above points. During the main analysis, AD+MCI group was set to the specific target of prediction. As a diagnostic application, the level of cognitive impairment need to be able to be estimated using features of PLRs if it is to be would be effective. If it were possible, AD and MCI should be classified using weighted levels.

The factor scores for blue but not red light pulses were selected once more for use in a regression model, following a stepwise procedure. The dominance of blue light pulses for the prediction was confirmed in a previous study [11], [12]. The possible reason for this may be based on the first hypothesis, which could not be confirmed according to the above evaluation, however. Therefore, a more detailed analysis needs to be conducted.

In this study, other metrics of cognitive functional ability have been observed such as VSRAD (Voxel-based Specific Regional analysis system for Alzheimer's Disease), HDS-R (Hasegawa's Dementia Scale-Revised) and MoCA-J (Japanese version of Montreal Cognitive Assessment). The development of an alternative diagnostic procedure which considers the level of cognitive functioning together with these metrics will be a subject of our further study.

V. SUMMARY

A procedure for detecting the level of cognitive impairment of senior citizens is examined using pupil light reflex (PLR) for chromatic light pulses and a portable measuring equipment. Features of PLRs are compared between blue and red light pulses.

- 1) PLRs are compared between left and right eyes when light pulse provides either eye. In addition, the latent factor scores of PLR features are also extracted. There are no significant differences in features and factor scores between the left and right eyes, however.
- 2) Factor scores and participant's ages were analyzed in order to classify individuals into groups such as AD+MCI and NC. Participant's age information contributed to classification of the groups. During the regression analysis using a variable selection procedure, factor scores for blue light pulses were extracted. PLRs for blue light pulses are key to accurate prediction.

A more accurate prediction procedure and method of analysis of the response mechanisms will be subjects of our further study.

ACKNOWLEDGEMENT

The authors would like to thank Prof. Masatoshi Takeda and Prof. Takenori Komatsu of Osaka Kawasaki Rehabilitation University, Toshinobu Takeda, MD at the Jinmeikai Clinic, Yasuhiro Ohta and Takato Uratani of the Uratani Lab Company Ltd. for their kind contributions,

REFERENCES

- [1] D. F. Fotiou, V. Setergiou, D. Tsiptsios, C. Lithari, M. Nakou, and A. Karlovasitou, "Cholinergic deficiency in Alzheimer's and Parkinson's disease: Evaluation with pupillometry," *International Journal of Psychophysiology*, vol. 73, pp. 143–149, 2009.
- [2] D. M. Bittner, I. Wieseler, H. Wilhelm, M. W. Riepe, and N. G. Müller, "Repetitive pupil light reflex: Potential marker in Alzheimer's disease?" *Journal of Alzheimer's Disease*, vol. 42, pp. 1469–1477, 2014.
- [3] J. K. H. Lim, Q.-X. Li, Z. He, A. J. Vingrys, V. H. Wong, N. Currier, J. Mullen, B. V. Bul, and C. T. O. Nguyen, "The eye as a biomarker for Alzheimer's disease," *Frontiers in Neurology*, vol. 10, no. 536, pp. 1–14, 2016.
- [4] S. Asanad, F. N. Ross-Cisneros, E. Barron, M. Nassisi, W. Sultan, R. Karanjia, and A. A. Sadun, "The retinal choroid as an oculo-vascular biomarker for Alzheimer's dementia: A histopathological study in severe disease," *Alzheimer's & Dementia: Diagnosis, Assessment & Disease Monitoring*, vol. 11, pp. 775–783, 2019.
- [5] P. D. Gamlin, D. H. McDougal, and J. Pokorny, "Human and macaque pupil responses driven by melanopsin-containing retinal ganglion cells," *Vision Research*, vol. 47, pp. 946–954, 2007.
- [6] A. Kawasaki and R. H. Kardon, "Intrinsically photosensitive retinal ganglion cells," *Journal of Neuro-Ophthalmology*, vol. 27, pp. 195–204, 2007.
- [7] A. J. Zele, P. Adhikari, D. Cao, and B. Feigl, "Melanopsin and cone photoreceptor inputs to the afferent pupil light response," *Frontiers in Neurology*, vol. 10, no. 529, pp. 1–9, 2019.
- [8] P. S. Chougule, R. P. Najjar, M. T. Finkelstein, N. Kandiah, and D. Milea, "Light-induced pupillary responses in Alzheimer's disease," *Frontiers in Neurology*, vol. 10, no. 360, pp. 1–12, 2019.
- [9] M. Nakayama, W. Nowak, H. Ishikawa, K. Asakawa, and Y. Ichibe, "Discovering irregular pupil light responses to chromatic stimuli using waveform shapes of pupillograms," *EURASIP J. in Bioinformatics and System Biology*, vol. #18, pp. 1–14, 2014.
- [10] A. J. Oh, G. Amore, W. Sultan, S. Asanad, J. C. Park, M. Romagnoli, C. L. Morgia, R. Karanjia, M. G. Harrington, and A. A. Sadun, "Pupillary evaluation of melanopsin retinal ganglion cell function and sleep-wake activity in pre-symptomatic Alzheimer's disease," *PLoS ONE*, vol. 14, no. 12, pp. 1–17, December 2019.
- [11] W. Nowak, M. Nakayama, T. Kręcicki, E. Trypka, A. Andrzejak, and A. Hachoł, "Analysis for extracted features of pupil light reflex to chromatic stimuli in Alzheimer's patients," *EAI Endorsed Transactions on Pervasive Health and Technology*, vol. 5, pp. 1–10, November 2019, e4.
- [12] W. Nowak, M. Nakayama, T. Kręcicki, and A. Hachoł, "Detection procedures for patients of Alzheimer's disease using waveform features of pupil light reflex in response to chromatic stimuli," *EAI Endorsed Transactions on Pervasive Health and Technology*, vol. 6, pp. 1–11, December 2020, e6.
- [13] W. Nowak, M. Nakayama, E. Trypka, and A. Zarowska, "Classification of Alzheimer's disease patients using metric of oculo-motors," in *Proceedings of the Federated Conference on Computer Science and Information Systems (FedCSIS)*, 2021, pp. 403–407.
- [14] D. H. McDougal and P. D. Gamlin, "Autonomic control of the eye," *Comprehensive Physiology*, vol. 5, no. 1, pp. 439–473, 2015.

Artificial Intelligence in Personalized Healthcare Analysis for Women’s’ Menstrual Health Disorders

Łukasz Sosnowski
 Systems Research Institute,
 Polish Academy of Sciences
 Nowelska 6, 01-447 Warsaw, Poland
 sosnowsl@ibspan.waw.pl

Joanna Żuławińska
 OvuFriend Sp. z o.o.
 Żłota 61/100,
 00-819 Warsaw, Poland
 joanna.zulawinska@ovufriend.com

Soma Dutta
 University of Warmia and Mazury
 in Olsztyn
 Słoneczna 54, 10-710 Olsztyn, Poland
 soma.dutta@matman.uwm.edu.pl

Iwona Szymusik
 Department of Obstetrics and Gynecology,
 Medical University of Warsaw,
 Zwirki and Wigury 61,
 02-091 Warsaw, Poland
 iwona.szymusik@gmail.com

Aleksandra Zyguła
 Department of Obstetrics and Gynecology,
 Medical University of Warsaw,
 Zwirki and Wigury 61,
 02-091 Warsaw, Poland
 am.zygula@gmail.com

Elżbieta Bambul-Mazurek
 Uniwersyteckie Centrum Zdrowia
 Kobiety i Noworodka WUM Sp. z o.o.
 Plac Starynkiewicza 1/3;
 02-015 Warsaw, Poland
 ela.bambul@gmail.com

Abstract—The paper presents an AI-based model which depending on the input of a woman for a finite number of menstrual cycles helps in determining the possible ovulation dates as well as possibility of some health risks e.g., Premenstrual Syndrome, Luteal Phase Defect etc. The architecture of the model consists of three layers, namely analyzing and detecting the features from a single cycle, analyzing cycle level concepts based on the analyzed features, and analyzing the user’s health risks based on the cycle level concepts accumulated over a finitely many cycles.

I. INTRODUCTION

IN THE last decades, in parallel to the industrial and social progress, several healthcare paradigms have appeared in the scientific medical community. These paradigms are proposing changes in the way in which healthcare is deployed in our society. The image of Traditional Medicine, where physicians are artists that are isolated and taking decisions based only on their knowledge and experience, are changing to a new doctor always connected and with access to the last evidence existing in a globalized world. [1]

As examples of new paradigms of medical treatment author of [1] refers to Evidence-Based Medicine [2], Personalized Medicine [3] etc. The key issue in all of them is to create the protocols and guidelines for medical care by combining the knowledge from the existing literature of medicine, experience of the professionals, as well as the input parameters, habits, life style, and preferences of the individual patients. However, implantation of Evidence-Based medicine and personalized medicine together in a platform of healthcare have some challenges; one way it needs standardization of protocols at least with respect to the consensus of a group, and on the other hand, it needs to be case sensitive by considering treatments

Co-financed by the EU Smart Growth Operational Program 2014-2020 under the project "Developing innovative solutions in the domain of detection of frequent intimate and hormonal health disorders in women of procreative age based on artificial intelligence and machine learning - OvuFriend 2.0", POIR.01.01.01-00-0826/20.

out of the guidelines in the context of the patients who have different responses to the standard treatment.

The concern of this paper is related to what has been mentioned in the paradigmatic change in medical treatment, in particular in the context of infertility [4] which has become a civilization disease. According to statistics every fifth couple, trying to conceive (TTC), has a problem to achieve pregnancy in the first 12 months of efforts, and this tendency is increasing [5]. Moreover, the age of women trying for the first child statistically shifts towards 35, which increases a risk in pregnancy, including the birth of a child with defects.

This paper is in continuation of a series of papers [4], [6]–[8] that attempted to establish a platform, known as OvuFriend 1.0¹, for helping women in determining the possibility of conceiving and understanding the hidden risk of getting related health problems based on their data input. The platform of OvuFriend 1.0 is provided as a mobile app where an user can put the data related to her physical and mental states during a specific menstrual cycle, and the underlying algorithm of the app helps to get an analysis of the possibility of conceiving or not conceiving. As mentioned in [8], OvuFriend 1.0, the result of a R&D project finished in 2020, brought the company OvuFriend a big commercial success because of its underlying AI algorithm [7] dedicated to the prediction and confirmation of ovulation supporting the natural endeavour for family planning methods [9].

As a continuation to the above mentioned achievement the second R&D project, called as OvuFriend 2.0, is aimed at extending the previous platform by adding the ability of analyzing and assessing the risk of having certain health disorders based on the given input of a woman. Identifying the increased risk will give a chance to refer to the right doctor and heal the ailment faster. In particular, the project

¹www.ovufriend.pl

focuses on the analysis of whether a particular user has the possibility of having the risk of Premenstrual Syndrome (PMS²), Luteal Phase Defect (LPD³), benign growths like polyps, fibroids⁴ in the uterus, Polycystic Ovary Syndrome (PCOS⁵) or hypothyroidism⁶. PMS is a combination of symptoms that many women get about a week or two before their period. Severe PMS symptoms may be a sign of premenstrual dysphoric disorder (PMDD). On the other hand, LPD is a health condition that may play a role in infertility. Fibroids and polyps too may cause infertility or recurrent pregnancy loss. This paper focuses on the schemes for detecting PMS, LPD, and other anatomical changes like polyp and fibroids.

The general scheme in OvuFriend 2.0 for having an AI based app determining the possible days of ovulation as well as the possibility of the above mentioned health risks goes to a great extent in the line of Evidence-based Medicine and Personalized Medicine. In particular, the following features, that are included in the model proposed by OvuFriend 2.0, strengthen the support for a Personalized Medicine.

It has three hierarchical levels, known as *Detector level*, *Cycle level*, and *User level*.

- (i) At the *detector's level* the user can put information related to her mental and physical health over one complete cycle. A set of attributes are chosen by the medical experts. Based on the provided input by a particular user the values for those attributes are determined by a team of medical experts and they are tagged against the information details of the patient. So, while preprocessing the data, the model aggregates the perception of the user as well as the knowledge and experience of a team of experts.
- (ii) Based on the values of the attributes from a completed cycle, certain compound concepts such as *ovulation happened*, *days of ovulation*, *follicular phase interval*, *luteal phase interval*, *PMS score* etc are determined. These are called *cycle-level concepts* and for determining such concepts the system is fed with some relevant formulas involving the attributes prefixed at the detector level. These formulas are formulated by abstracting relationships among different attributes as described by a team of medical experts based on their knowledge from the literature and personal experiences. So, in the proposed model the mathematical formulations of the interrelationships among different attribute values are discovered by aggregating a team of medical experts' opinions.
- (iii) In the third level, the system aggregates the data related to

²<https://www.womenshealth.gov/menstrual-cycle/premenstrual-syndrome>

³<https://www.webmd.com/infertility-and-reproduction/guide/luteal-phase-defect>

⁴<https://progyny.com/education/female-infertility/understanding-uterine-fibroids-polyps/>

⁵In this condition the ovaries produce an abnormal amount of androgens, that are usually present in women in small amounts [10].

⁶Hypothyroidism means the thyroid gland does not produce enough thyroid hormones, which can lead to changes in the menstrual cycle. (<https://helloclue.com/articles/cycle-a-z/hypothyroidism-and-the-menstrual-cycle>)

the detector level as well as the cycle level concepts of a particular user for a finitely many cycles. This level focusing on the user's history is known as the *user's level*. The examples of the user level concepts are *risk of PMS*, *risk of LPD*, *risk of infertility* etc. Here, the system calculates the probabilistic ratio of the above mentioned cycle level concepts over the total number of cycles considered for a particular user. Moreover, the system is also fed with a threshold value for each such user level concepts and these threshold values are learned or even adjusted based on the opinions of the medical experts and the histories of already recorded and analysed cases. If the respective ratio for a particular user level concept is greater than the prefixed threshold for that concept the system notifies the user about the possibility of such health risk. So, at this level the threshold, chosen for a particular health risk, is set based on both experts' knowledge and current existing evidences of such cases.

The above discussed general scheme is presented in Fig. 1. The process of determining ovulation was described in the previous publication as part of the scope of the previous project (Ovufriend 1.0) [7].

Thus, as a whole the model endorses a three-layered hierarchical learning and reasoning mechanism based on the knowledge and experiences of a team of medical experts, perceptions of the users, and already recorded evidences to the system. Furthermore, the hierarchy of approximating fuzzy concepts is developed by using the quantifiers of fuzzy linguistic summaries in the process of inferring and making local decisions [11], [12]. From this angle, the model designed in OvuFriend 2.0 complies to a great extent to the need of personalized and evidence based medicine. On the other hand, the model also endorses some features of Interactive Granular Computing (IGrC) [13], [14], by incorporating perception of the current health situation of a woman based on the individual spatio-temporal windows of the physical world and actual physical interactions in the form of measuring attributes in the given space and time windows.

The content of the chapter is organized as follows. Section II presents the development made under OvuFriend 2.0; it is divided into several subsections describing the schemes for determining PMS, analysing the risk of LPD, and indicating anatomical changes related to polyp, fibroids etc. Further in section III, the reference set, used for experiments, is described, and the obtained results are explained in section IV. The paper ends with a concluding section indicating future directions of developing the model.

II. AI ALGORITHMS DETERMINING WOMEN HEALTHCARE RISKS

In this section we would present the framework of OvuFriend 2.0 by describing the AI algorithms and schemes for determining whether an user has the risks of certain health diseases. Specifically, we focus on the health diseases such as PMS, LPD, Fibroids and Polyps. All these schemes are discussed below in separate subsections.

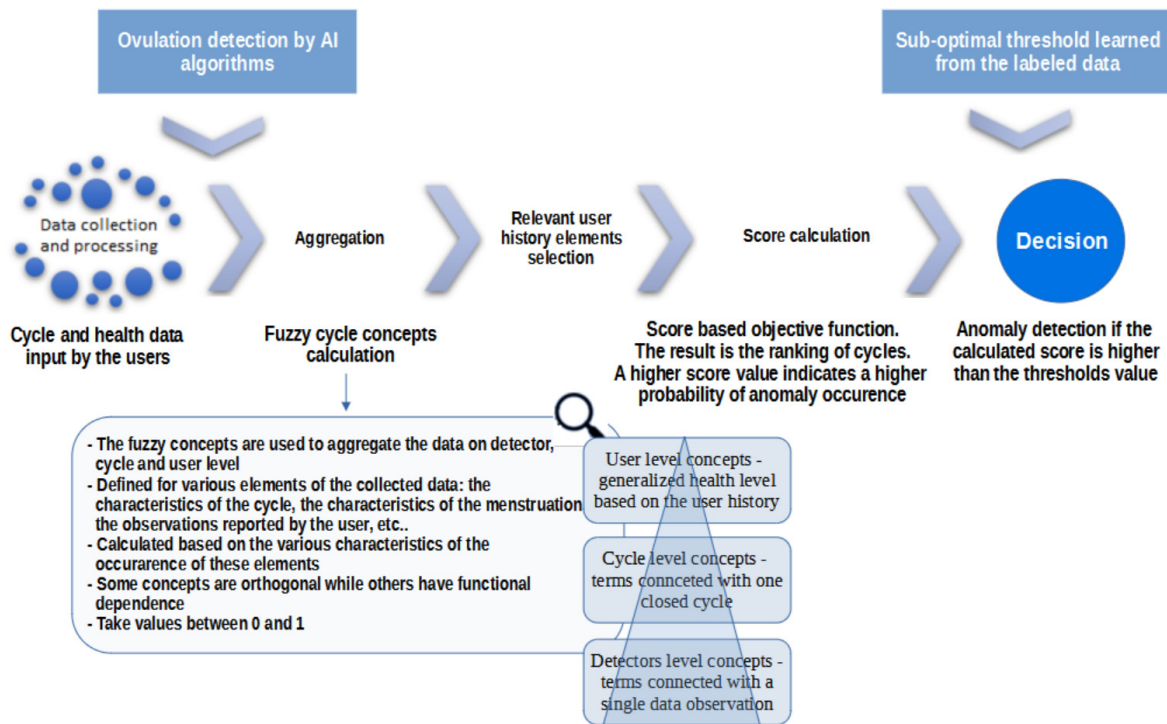


Fig. 1. General scheme for OvuFriend 2.0

A. Scheme to determine risk for PMS

The prerequisite to start this scheme is to collect data related to the physical and mental health of a woman before, during, and after a complete menstrual cycle. After the completion of a cycle, with the gathered data, analysis for the risk of PMS starts. Initially, the data is processed to investigate whether the ovulation has occurred and whether it is possible to determine the day of its occurrence. At this stage all concepts pertaining to the detector level are analysed and determined. For example, if ovulation has been determined, an attempt is made to indicate two intervals of equal length falling into the follicular phase and the luteal phase of the cycle respectively.

The length of the intervals depends on the length of menstruation, the day of ovulation, and the length of the total cycle. A complete cycle means number of days between starting of the menstruation in one month to the starting of the same in the next month. The beginning point of the first interval is chosen as the k -th day after the end of the monthly menstruation of the current month, where the value for k is prefixed in the algorithm. If the length of cycle is x and number of days of the current month menstruation is m , then each interval has to be of length $\frac{x-(k+m)}{2}$ and hence the beginning point and the end point of the first interval are respectively $m+k$ and $\frac{x+(k+m)}{2}$. Consequently, the beginning point of the second interval is $\frac{x+(k+m)}{2}$ and the the end point is x , the last day of the cycle.

Now, if the intervals are successfully determined, the coefficients of occurrence of the physical symptoms and mood

symptoms characteristics of PMS are calculated. The set of mood symptoms is presented in the Fig. 5. This set of symptoms and formulas for calculating the coefficients based on them are defined based on the interactions with a team of medical experts and aggregating their consensus of gathered knowledge and experiences about variations of different moods, feelings and physical impacts observed in women during the menstrual cycles. Each such symptom related to physical or mood aspects is counted and it is checked whether they occur in both the phases or only in the second phase. If there is at least one physical symptom or mood symptom that occurs in both phases, the algorithm reduces the weights in the respective formula calculating the mood feel coefficient or physical feel coefficient. Usually the physical or mood impacts during the second phase are only reflected in the occurrences of PMS, and that is why, while some symptoms are observed in both the phases, the possibility for PMS is decreased by reducing the weights. Finally, by aggregating the number of physical symptoms and the mood symptoms in a particular phase the coefficients are calculated separately for the physical symptoms and the mood symptoms according to the following formulas. Fig. 2 shows the algorithmic flowchart behind the described process for determining the cycle level concept *PMS score*, denoted as PMS_{score} .

Let us denote the two phases as P_1 and P_2 respectively.

$$P_iMoodFeelCoeff = \frac{(SumOfOccurrenceP_iMood)}{K_1 \times PhaseLength} \times \alpha + (1-\alpha) \tag{1}$$

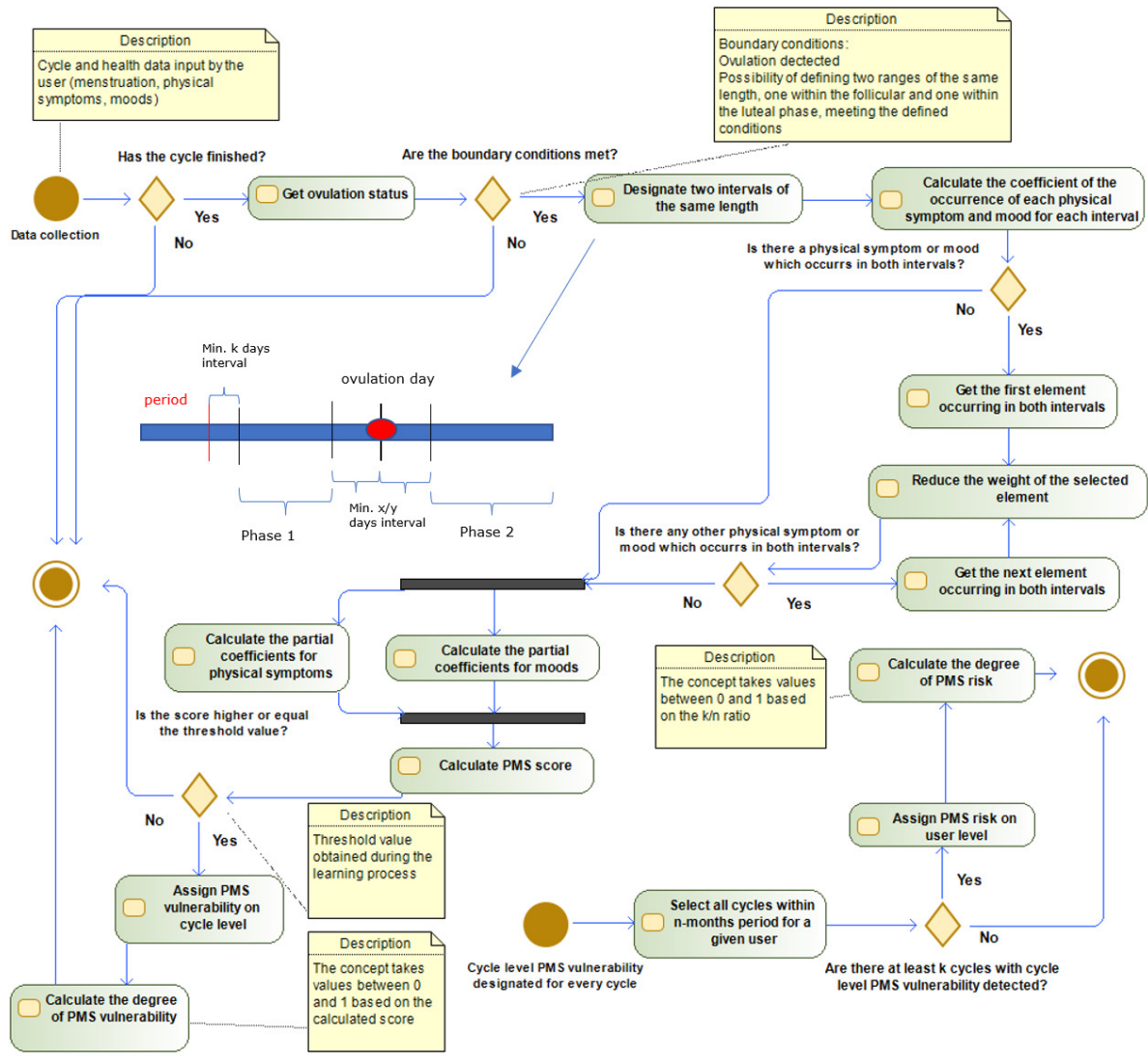


Fig. 2. Complete scheme for calculating PMS risk vulnerability (2 diagrams)

where $i = 1, 2$ and $\alpha \in (0, 1)$,

$$P_2PhysFeelCoeff = \frac{(SumOfOccurrenceP_2Phys)}{K_2 \times PhaseLength} \times \beta + (1-\beta) \quad (2)$$

where $\beta \in (0, 1)$.

The symbols $SumOfOccurrenceP_iMood$ and $SumOfOccurrenceP_iPhys$, used in the above equations, respectively indicate the number of mood and the number of physical symptoms occurred in a particular phase P_i . The symbols K_1 and K_2 represent respectively the total number of all moods and physical symptoms listed in the system. The factors α and β are parameters to control the significance of the given components in the final calculation of the result. The formulas are designed by a team of scientific experts based on the general description given by the medical experts regarding the effect on physical and mental health of women during a cycle as well as keeping into account the observed

patterns of cases available in the record. Based on the above coefficients the cycle-level concept, namely $PMS\ score$ is calculated in accordance to the following formula.

$$PMS_{score} = \frac{P_2MoodFeelCoeff}{P_1MoodFeelCoeff} + \frac{P_2PhysFeelCoeff}{w_1} \quad (3)$$

where w_1 is the weight chosen by a team of medical experts.

At the beginning, the algorithm also fixes a threshold for PMS score by taking into account already available records of patients. The threshold may vary over time based on the changes in the patients' record, and thus, in some sense the underlying algorithm keeps a possibility of learning the threshold for PMS score based on the current evidences. Based on this threshold whether an user has the PMS susceptibility or not is determined just by checking if the obtained score is greater or equal to the prefixed threshold. If during the current

cycle the algorithm determines PMS susceptibility for an user the algorithm passes to the next level where the degree of PMS risk is calculated for a particular user based on the observations of finitely many cycles.

The cycle level compliance data is used to investigate PMS risk at the user level. If over the selected period of n months there are at least k cycles with PMS susceptibility, then the PMS risk is assigned to the user, and its degree is calculated simply by the value of $\frac{k}{n}$ in the range $[0, 1]$. The flowchart, presented in the Figure 2, shows a complete overview of the algorithm specifying PMS susceptibility and PMS risk for a particular user.

B. Scheme to determine risk for LPD

The general prerequisite for running the algorithm to determine the risk of Luteal Phase Defect (LPD) [15] is common to all considered disorders but differs in details. At the beginning stage, preprocessing of the input data and analyzing the detector level concepts such as *whether ovulation has occurred* are performed, and then based on that the boundary conditions are calculated. These conditions are verified using fuzzy quantifiers of linguistic summaries operating on multivariate time series (e.g., the quantifier *exists*) [16]. The specific scheme of LPD differs from that of PMS in the formula that is fed to the algorithm in order to calculate the susceptibility of LPD and then consequently its degree of risk.

Similar like, PMS score, here the AI algorithm is fed with a formula for calculating LPD score, given by the following equation.

$$LPD_{score} = w_1 * LutParameters + w_2 * DecFer \quad (4)$$

Here both $LutParameters, DecFer \in [0, 1]$ and they respectively denote the values for *Luteal Phase Parameters* and *Decreased Fertility*. The *Luteal Phase Parameters* are determined based on the luteal phase length and various other factors related to the analysis of bleeding during the luteal phase. On the other hand, the *Decreased Fertility* concerns about the observation of the period of time in which the attempts are made for conceiving a child, the number of miscarriages (they are estimated on the basis of pregnancy tests performed and the length of the cycle compared to the typical lengths of the cycle and luteal phase for a given user), etc. Both the respective values for $LutParameters, DecFer$ are obtained based on the input data of the particular user; whereas w_1, w_2 are some weights that are chosen by the team of experts based on their collective knowledge regarding which of $LutParameters$ and $DecFer$ are significant to what extent.

Once in the cycle level the algorithm determines the possibility of LPD, it passes to the next level and as in the case of PMS risk the algorithm calculates the degree of risk for LPD; that is, if in the selected period of n months there are at least k cycles with LPD susceptibility, then the LPD risk is simply the value of $\frac{k}{n}$. For an overview of the whole scheme the readers are referred to Fig. 3.

C. Scheme for indicating anatomical changes like polyps and fibroids

As in the cases for PMS and LPD, here also the analysis for the presence of anatomical anomalies starts with the data of a complete cycle of an user. The primary analysis is manifested by focusing on the data related to inter-menstrual bleedings or spots. As usual, initially, the data is processed to investigate whether the ovulation has occurred and whether it is possible to designate a possible ovulation date. At the same time, the detector level concepts are also determined and then cycle label concepts are analyzed in the same fashion as mentioned in the cases of PMS and LPD.

If the required data is obtained during a complete cycle so that the algorithm becomes able to determine the occurrence of ovulation or an-ovulation, the process proceeds to the next stage of the examination of the disease. The cycle level concepts, which are associated to the symptoms characterizing the particular diseases like polyp or fibroids, are selected. On their basis, a score is calculated in accordance to the formula presented below.

$$Score = w_1 * DisMens + w_2 * DecFer + w_3 * PhysSymp \quad (5)$$

Here all the weights w_1, w_2, w_3 are chosen by the team of experts, and $DisMens$, the value for the parameters corresponding to disordered menstruation, $DecFer$, the value for decreased fertility, and $PhysSymp$, the values corresponding to physical symptoms related to such diseases, are obtained from the input data of a particular user. All these values are scaled in the interval $[0, 1]$ based on the information related to inter-menstrual bleeding, long-lasting menstruation, intensity of menstruation, miscarriage, long trying time for conceiving, pelvis pain, polyuria etc.

As before, in this context also if a cycle's score is greater than or equal to the cut-off value, which is set through some learning process, the cycle is assigned anatomical susceptibility at that particular cycle level. Then its grade is calculated in the range of $[0, 1]$ depending on out of n cycles in how many cycles the algorithm agrees with the susceptibility of anatomical changes occurred in the case of a particular user; in other words, it is simply $\frac{k}{n}$ if in k such cycles susceptibility of anatomical changes is detected.

The full scheme for determining the possibility of such diseases as polyp, fibroids, can be visualized in Fig. 4.

III. REFERENCE SET BASED ON EXPERTS KNOWLEDGE

In order to estimate the effectiveness of the algorithms detecting anomalies, a reference set has been created consisting of cycles described by the experts. For each cycle the experts are provided the information e.g., the chance of an anomaly occurrence expressed by a value in the range $[0, 1]$ and a comment explaining the assessment made. They are also provided with a cycle visualization containing the basic data needed to determine the ovulation, as well as the information of already predicted ovulation (product of the Ovufriend 1.0 project). In addition, the visualization contains

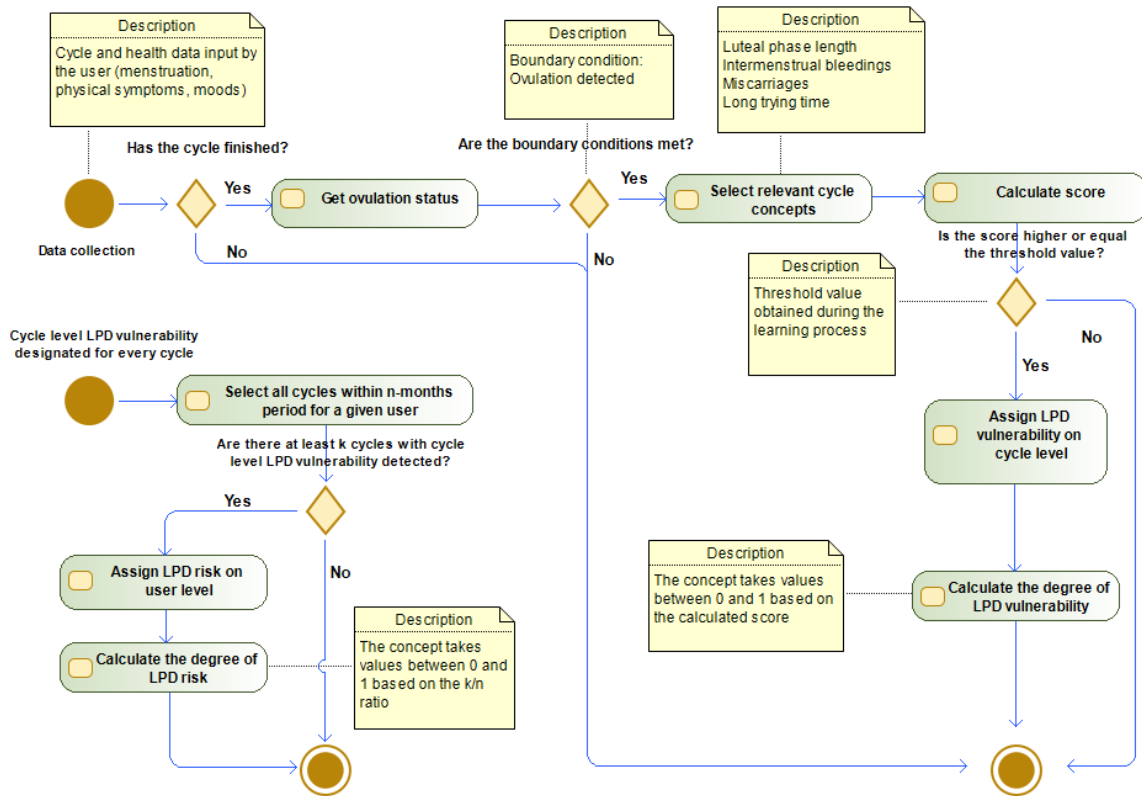


Fig. 3. Complete scheme for LPD risk vulnerability (2 diagrams)

a series of low level data (e.g., group of symptoms) broken down by observation days. As a whole it can be considered as a multidimensional time series indexed with the days of the ovulatory cycle. An example of a visualization labelling form for *PMS* can be found in Fig. 5 [17], and the same for luteal phase deficiency (LPD) can be found in Fig. 6. The form for tagging polyps and fibroids are similar to the one for *LPD*. It is extended with a few additional attributes and it is shown in Fig. 6. The reference set consists of 900 menstrual cycles, and in the category of "Anatomical or hormonal abnormalities with inter-menstrual spots" each cycle is tagged for both the presence of NFL and anatomical changes. The number of items in the set has grown steadily as successive sets of cycles are submitted for tagging. Subsequent sets of cycles are drawn

TABLE I
THE SAMPLE SIZE BY CLASS AND TYPE OF ANOMALY. PC - POSITIVE CLASS, NC - NEGATIVE CLASS

Anomaly	#	PC	% PC	NC	% NC
PMS	300	164	55%	136	45%
LPD	300	147	49%	153	51%
Polyps and Fibroids	300	160	53%	140	47%

on the basis of the given criteria, selected in such a way that it helps to obtain a similar size in the positive class (minimum score 0.5) and the negative class (score below 0.5). The size

of each class, broken down by a group and the types of anomaly, is presented in Table I. In each of the three groups of anomalies, the sample is well-balanced, where the share of the positive class ranges from 49% to 55% of the sample.

IV. EXPERIMENTS AND RESULTS

In order to evaluate the effectiveness of the prototypes of the algorithms, four experiments have been conducted for each of the three disorders. Overall the methodology looks as follows. Each of the experiments involves a different number of repetitions such as 1000, 500, 100, and 10, respectively for ReSample evaluation [18]. Each time 33% of cases from the reference set are drawn. Training is performed on the selected subset, and testing is performed on the remaining 67%. On each iteration, the intersection of both sets remains empty. The values of the cut-off thresholds of rankings for all described disorders are learned in each iteration from training set that consists of 100 cycles (33% of 300 tagged cycles). The test procedure that returned the values for contingency table are processed on 200 cycles each time. Single iteration results are stored in the contingency table. Finally, all TP (True Positive), TN (True Negative), FP (False Positive) and FN (False Negative) are summed and then measures of effectiveness are calculated. The obtained results for 1000 repetitions are presented in the Table II.

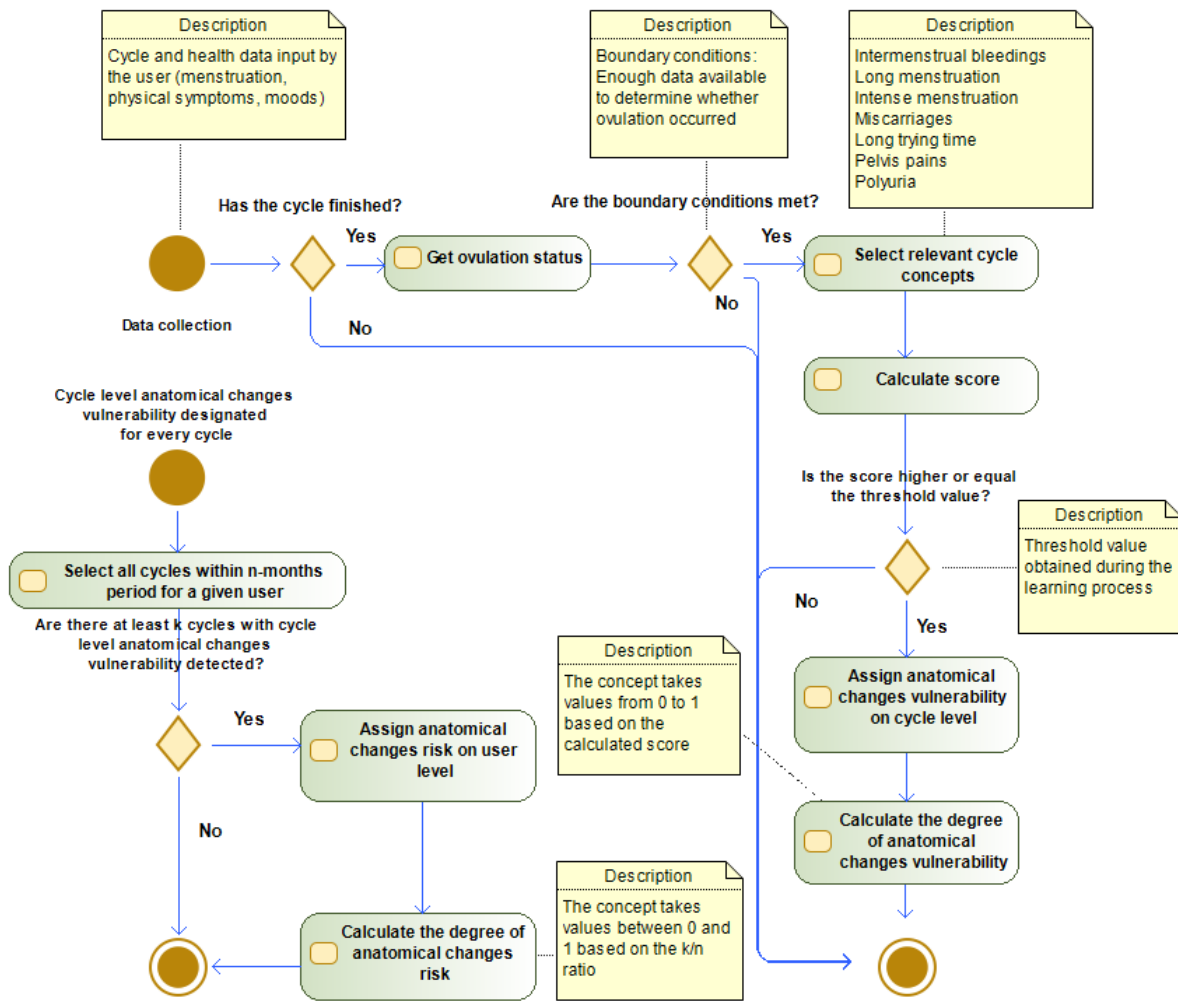


Fig. 4. Complete scheme for determining risk vulnerability of polyps and fibroids (2 diagrams)

The case for TP is assigned when the cycle is tagged with at least 0.5 by the medical experts and the algorithm has calculated the score that positioning the cycle in the positive set of a given disease; on the other hand, the case for FP is assigned when the experts have given mark below 0.5 but the algorithm has classified the case into positive class. The case for TN is obtained when the experts have assigned less than 0.5 and the algorithm has calculated score under the learned threshold. Finally the cases for FN is indicated when the algorithm has calculated the score value under the learned threshold but from the experts it receives a mark greater or equal 0.5. The evaluation results for 500 repetitions are presented in the Table III.

The obtained results for 100 and 10 repetitions are presented in the Table IV and Table V respectively.

In the project, the team of medical experts consists of three

highly qualified medical scientists, and the decisions regarding tagging cycles have been made based on discussion within the team. As the values, selected after tagging, are already agreed with the consensus of the whole team they do not require additional processing in order to be used for evaluation.

The presented results of the experiments concern the first stage of the project, in which the parameters of the algorithms are learned from the tagging of medical experts. The labels were made on the data constituting the subjective observations of the system users and the measurements of the physical parameters (e.g., BBT, cervical mucus, cervix position, etc.) and the subjective labels of the team of experts. In the next stage, the data and the final evaluation will be extended on the real test results, that are used by the doctors to make the diagnosis. In this way, the algorithm will be tested against the real hardcore medical data.

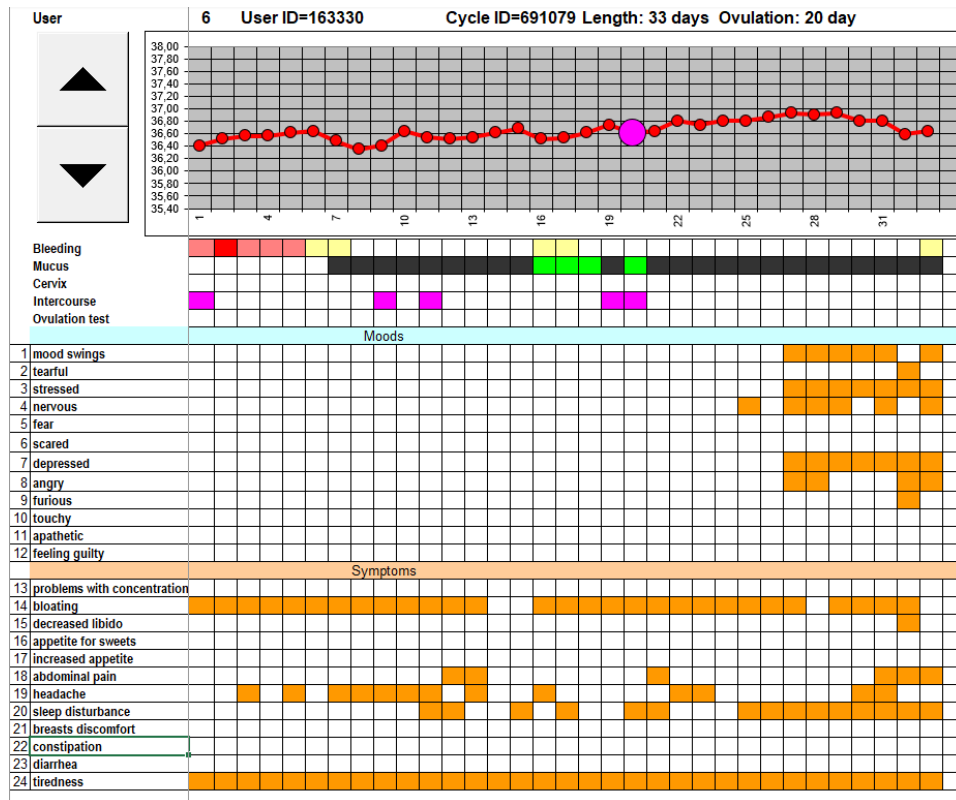


Fig. 5. PMS labelling form prepared for medical experts to evaluate susceptibility of selected cycles. Form is based on such attributes as: bleeding, mucus, bbt, cervix, mood swings, tearful, stressed, nervous, depressed, angry, furious, tiredness, bloating, problem with the concentration, appetite for sweets, sleep disturbance, breasts pains, constipation, etc.

TABLE II

RESULTS AVERAGED OVER 1000 ITERATIONS OF THE ReSAMPLE ROUTINE. ABBREVIATIONS: # - SAMPLE, TP - TRUE POSITIVES, TN - TRUE NEGATIVES, FP - FALSE POSITIVES, FN - FALSE NEGATIVES, PR - PRECISION, RE - RECALL, F1 - F1 SCORE, ACC - ACCURACY, POL-FIBR - POLYPS AND FIBROIDS

Type	#	TP	TN	FP	FN	PR	RE	F1	ACC	min_F1	max_F1	min_ACC	max_ACC
LPD	200000	88650	87920	13991	9439	0.864	0.904	0.883	0.883	0.781	0.925	0.795	0.920
PMS	200000	96685	70710	19921	12684	0.829	0.884	0.856	0.837	0.715	0.900	0.725	0.885
POL-FIBR	200000	86714	82157	10990	20139	0.888	0.812	0.848	0.844	0.731	0.900	0.760	0.895

V. CONCLUSION

AI and machine learning based techniques are nowadays prevalent in every sphere of life and the healthcare industry is also not out of that influence of automated decision support. In the Introduction, the terms like personalized medicine and evidence based medicine are presented in the context of the need for a new paradigm of medical treatment. It is expected that in the new context, treating a patient should not be an isolated process conducted by an individual doctor based on his/her knowledge and experience about a particular field of medicine. Moreover, there should be a standardization in the process of treating a particular disease by different doctors.

In this regard, the attempt of OvuFriend 2.0 has been to develop an AI-based model for women health care where based on the input of a particular user the model can suggest the possibility of certain health risks. The architecture of the model is developed in such a way that the system has an

interface of user in order to gather input data as well as an interface of a team of medical experts who based on a consensus creates a protocol for standardizing lowest level concepts, known as detector level concepts, and determining their values. Based on a complete cycle of data and values selected for detector level concepts the next level concepts, known as cycle level concepts, are analyzed and evaluated. Finally, at the highest level, known as user level, the degree of risk for certain cycle level concepts are computed by considering the values obtained for those concepts for a finitely many cycles.

Firstly, the user interface of the model keeps it sensitive to the user's perceptions and thus incorporates the aim of personalized medicine. Secondly, the interface for a team of medical experts keeps the possibility open for discussion, standardization, and revision of the defining criterion for medical concepts, and at the same time the process of treatment

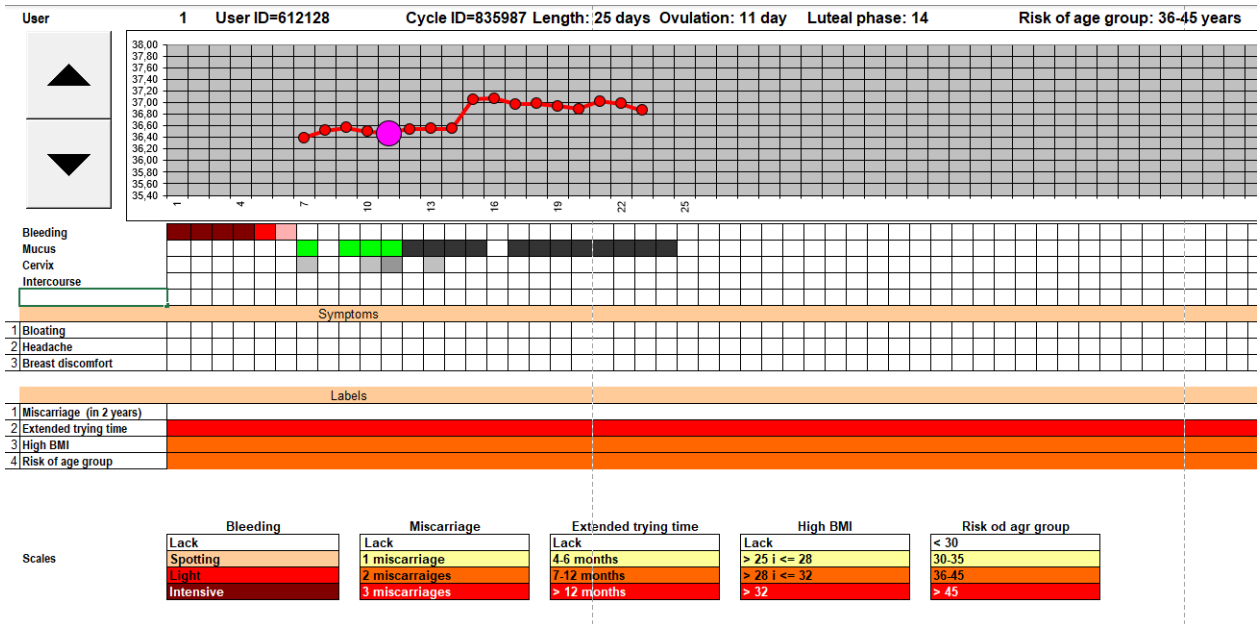


Fig. 6. Luteal phase deficiency and polyps tagging form prepared for medical experts to evaluate susceptibility of selected cycles. Form is based on such attributes as: bleeding, mucus, bbt, cervix, age group, increased BMI, extended trying time, miscarriages in history, etc.

TABLE III

RESULTS AVERAGED OVER 500 ITERATIONS OF THE ReSAMPLE ROUTINE. ABBREVIATIONS: # - SAMPLE, TP - TRUE POSITIVES, TN - TRUE NEGATIVES, FP - FALSE POSITIVES, FN - FALSE NEGATIVES, PR - PRECISION, RE - RECALL, F1 - F1 SCORE, ACC - ACCURACY, POL-FIBR - POLYPS AND FIBROIDS

Type	#	TP	TN	FP	FN	PR	RE	F1	ACC	min_F1	max_F1	min_ACC	max_ACC
LPD	100000	44317	44071	7035	4577	0.863	0.906	0.884	0.884	0.789	0.921	0.805	0.915
PMS	100000	48043	35454	9965	6538	0.828	0.880	0.853	0.835	0.754	0.902	0.745	0.885
POL-FIBR	100000	43523	41078	5618	9781	0.886	0.817	0.850	0.846	0.747	0.910	0.770	0.905

TABLE IV

RESULTS AVERAGED OVER 100 ITERATIONS OF THE ReSAMPLE ROUTINE. ABBREVIATIONS: # - SAMPLE, TP - TRUE POSITIVES, TN - TRUE NEGATIVES, FP - FALSE POSITIVES, FN - FALSE NEGATIVES, PR - PRECISION, RE - RECALL, F1 - F1 SCORE, ACC - ACCURACY, POL-FIBR - POLYPS AND FIBROIDS

Type	#	TP	TN	FP	FN	PR	RE	F1	ACC	min_F1	max_F1	min_ACC	max_ACC
LPD	20000	8808	8833	1394	965	0.863	0.901	0.882	0.882	0.783	0.921	0.795	0.915
PMS	20000	9687	7025	2049	1239	0.825	0.887	0.855	0.836	0.760	0.911	0.760	0.895
POL-FIBR	20000	8672	8245	1122	1961	0.885	0.816	0.849	0.846	0.751	0.900	0.775	0.890

does not remain in the hand of one individual. Thirdly, the underlying AI algorithm has a updating mechanism which with time changes certain thresholds for analyzing health risks based on the already available records of the patients. Thus, the model incorporates a learning mechanism as well as supports the idea of evidence based medicine.

In the present version of the model, there are some aspects where lie the possibility of extension and improvement. Let us list them as immediate directions for future research.

- One of them is to make the two above mentioned interfaces interactive by introducing a language of dialogue [19], [20] so that the underlying treatment protocol can be updated or revised based on even a particular user's input. In the present model, certain weights for a particular health disease does not depend on the input of a

particular user. Incorporating this direction may make the model more dynamic in learning the optimized care for a particular user.

- In the present context, the formulas determining the values for the cycle level concepts and the user level concepts are fixed. In this context also there is a possibility of using machine learning techniques in order to learn a set of possible rules or formulas for diagnosing certain health risks based on the already existing evidences and making the process of diagnosing more flexible and evidence driven.

The current state of the algorithms shows a very good quality of the results achieved by the algorithms, while the real test will be their modernization and incorporation of data from medical examinations into operation. It will be a

TABLE V

RESULTS AVERAGED OVER 100 ITERATIONS OF THE RESAMPLE ROUTINE. ABBREVIATIONS: # - SAMPLE, TP - TRUE POSITIVES, TN - TRUE NEGATIVES, FP - FALSE POSITIVES, FN - FALSE NEGATIVES, PR - PRECISION, RE - RECALL, F1 - F1 SCORE, ACC - ACCURACY, POL-FIBR - POLYPS AND FIBROIDS

Type	#	TP	TN	FP	FN	PR	RE	F1	ACC	min_F1	max_F1	min_ACC	max_ACC
LPD	2000	902	873	120	105	0.883	0.896	0.889	0.888	0.845	0.924	0.855	0.920
PMS	2000	950	709	209	132	0.820	0.878	0.848	0.830	0.831	0.869	0.815	0.845
POL-FIBR	2000	868	824	115	193	0.883	0.818	0.849	0.846	0.791	0.877	0.810	0.875

milestone which, if achieved, will guarantee another success for applications and system users.

REFERENCES

- [1] Carlos Fernández-Llatas and Jorge Muñoz-Gama *et al.* "Process Mining in Healthcare", pages 41–52. Springer, 2020.
- [2] Brian Haynes and Andrew Haines. "Barriers and bridges to evidence based clinical practice". *BMJ*, 317(7153):273–276, 1998.
- [3] Margaret A. Hamburg and Francis S. Collins. "The Path to Personalized Medicine". *New England Journal of Medicine*, 363(4):301–304, 2010.
- [4] Lukasz Sosnowski and Tomasz Penza. "Generating Fuzzy Linguistic Summaries for Menstrual Cycles". volume 21 of *Annals of Computer Science and Information Systems*, pages 119–128, 2020.
- [5] L Bablok, W Dziadecki, I Szymusik, and *et al.* "Patterns of infertility in Poland - multicenter study". *Neuro Endocrinol Lett.*, 32(6):799–804, 2011.
- [6] Joanna Fedorowicz, Lukasz Sosnowski, Dominik Slezak, Iwona Szymusik, Wojciech Chaber, Lukasz Milobedzki, Tomasz Penza, Jadwiga Sosnowska, Katarzyna Wójcicka, and Karol Zaleski. "Multivariate Ovulation Window Detection at Ovufriend". In Tamás Mihálydeák, Fan Min, Guoyin Wang, Mohua Banerjee, Ivo Düntsch, Zbigniew Suraj, and Davide Ciucci, editors, *Rough Sets - International Joint Conference, IJCRS 2019, Debrecen, Hungary, June 17-21, 2019, Proceedings*, volume 11499 of *Lecture Notes in Computer Science*, pages 395–408. Springer, 2019.
- [7] Lukasz Sosnowski, Iwona Szymusik, and Tomasz Penza. "Network of Fuzzy Comparators for Ovulation Window Prediction". volume 1239 of *Communications in Computer and Information Science*, pages 800–813. Springer, 2020.
- [8] Lukasz Sosnowski and Jakub Wróblewski. "Toward automatic assessment of a risk of women's health disorders based on ontology decision models and menstrual cycle analysis". In Yixin Chen, Heiko Ludwig, Yicheng Tu, Usama M. Fayyad, Xingquan Zhu, Xiaohua Hu, Suren Byna, Xiong Liu, Jianping Zhang, Shirui Pan, Vagelis Papalexakis, Jianwu Wang, Alfredo Cuzzocrea, and Carlos Ordonez, editors, *2021 IEEE International Conference on Big Data (Big Data), Orlando, FL, USA, December 15-18, 2021*, pages 5544–5552. IEEE, 2021.
- [9] B Smoley and C Robinson. "Natural family planning". *Am Fam Physician*, 86(10):924–928, 2012.
- [10] Neil F. Goodman, Rhoda H. Cobin, Walter Futterweit, Jennifer S. Glueck, Richard S. Legro, and Enrico Carmina. "American Association of Clinical Endocrinologists, American College of Endocrinology, and Androgen Excess and PCOS Society Disease State Clinical Review: Guide to the Best Practices in the Evaluation and Treatment of Polycystic Ovary Syndrome - Part 1". *Endocrine Practice*, 21(11):1291–1300, 2015.
- [11] Janusz Kacprzyk, Ronald R. Yager, and José M. Merigó. "Towards Human-Centric Aggregation via Ordered Weighted Aggregation Operators and Linguistic Data Summaries: A New Perspective on Zadeh's Inspirations". *IEEE Comput. Intell. Mag.*, 14(1):16–30, 2019.
- [12] Janusz Kacprzyk and Sławomir Zadrozny. "Fuzzy logic-based linguistic summaries of time series: a powerful tool for discovering knowledge on time varying processes and systems under imprecision". *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, 6(1).
- [13] Soma Dutta and Andrzej Skowron. "Toward a Computing Model Dealing with Complex Phenomena: Interactive Granular Computing". In Ngoc Thanh Nguyen, Lazaros Iliadis, Ilias Maglogiannis, and Bogdan Trawinski, editors, *Computational Collective Intelligence - 13th International Conference, ICCCI 2021, Rhodes, Greece, September 29 - October 1, 2021, Proceedings*, volume 12876 of *Lecture Notes in Computer Science*, pages 199–214. Springer, 2021.
- [14] Andrzej Jankowski, Andrzej Skowron, and Roman W. Swiniarski. "Interactive Complex Granules". *Fundam. Informaticae*, 133(2-3):181–196, 2014.
- [15] Kenneth A. Ginsburg. "Luteal Phase Defect: Etiology, Diagnosis, and Management". *Endocrinology and Metabolism Clinics of North America*, 21(1):85–104, 1992. Reproductive Endocrinology.
- [16] Janusz Kacprzyk, Jan W. Owsinski, Eulalia Szmidi, and Sławomir Zadrozny. "Fuzzy Linguistic Summaries for Human Centric Analyses of Sustainable Development Goals (SDG) Related to Technological Innovations". In José L. Verdegay, Julio Brito, and Carlos Cruz, editors, *Computational Intelligence Methodologies Applied to Sustainable Development Goals*, volume 1036 of *Studies in Computational Intelligence*, pages 19–35. Springer, 2022.
- [17] Mehri Kalhor, Samaneh Yusefloo, Behroz Kaveii, Fatemeh Mohammadi, and Homa and Javadi. "Effect of Yarrow (*Achillea millefolium* L.) Extract on Premenstrual Syndrome in Female Students Living in Dormitory of Qazvin University of Medical Sciences". *Journal of Medicinal Plants*, 18(72), 2019.
- [18] P.I. Good. "Resampling Methods: A Practical Guide to Data Analysis". Birkhäuser Boston, 2005.
- [19] Soma Dutta and Andrzej Skowron. "Concepts Approximation Through Dialogue With User". volume LNCS 11499 of *IJCRS 2019*, pages 295–311, 2019.
- [20] S. Dutta and P. Wasilewski. "Dialogue in Hierarchical Concept Learning using Prototypes and Counterexamples". *Fundamenta Informaticae*, 162:17–36, 2018.

17th Conference on Information Systems Management

THIS event constitutes a forum for the exchange of ideas for practitioners and theorists working in the broad area of information systems management in organizations. The conference invites papers coming from three complimentary directions: management of information systems in an organization, uses of information systems to empower managers, and information systems for sustainable development. The conference is interested in all aspects of planning, organizing, resourcing, coordinating, controlling and leading the management function to ensure a smooth operation of information systems in organizations. Moreover, the papers that discuss the uses of information systems and information technology to automate or otherwise facilitate the management function are specifically welcome. Papers about the influence of information systems on sustainability are also expected.

TOPICS

- Management of Information Systems in an Organization:
 - Modern IT project management methods
 - User-oriented project management methods
 - Business Process Management in project management
 - Managing global systems
 - Influence of Enterprise Architecture on management
 - Effectiveness of information systems
 - Efficiency of information systems
 - Security of information systems
 - Privacy consideration of information systems
 - Mobile digital platforms for information systems management
 - Cloud computing for information systems management
- Uses of Information Systems to Empower Managers
 - Achieving alignment of business and information technology
 - Assessing business value of information systems
 - Risk factors in information systems projects
 - IT governance
 - Sourcing, selecting and delivering information systems
 - Planning and organizing information systems
 - Staffing information systems
 - Coordinating information systems
 - Controlling and monitoring information systems
 - Formation of business policies for information systems
- Portfolio management,
- CIO and information systems management roles
- Information Systems for Sustainability
 - Sustainable business models, financial sustainability, sustainable marketing
 - Qualitative and quantitative approaches to digital sustainability
 - Decision support methods for sustainable management

TECHNICAL SESSION CHAIRS

- **Arogyaswami, Bernard**, Le Moyne University, USA
- **Chmielarz, Witold**, University of Warsaw, Poland
- **Jankowski, Jarosław**, West Pomeranian University of Technology in Szczecin, Poland
- **Kisielnicki, Jerzy**, University of Warsaw, Poland
- **Ziemia, Ewa**, University of Economics in Katowice, Poland

PROGRAM COMMITTEE

- **Bicevska, Zane**, University of Latvia, Latvia
- **Bicevskis, Janis**, University of Latvia, Latvia
- **Carchiolo, Vincenzo**, Universita di Catania, Italy
- **Czarnacka-Chrobot, Beata**, Warsaw School of Economics, Poland
- **De Juana-Espinosa, Susana**, Universidad de Alicante, Spain
- **Duan, Yanqing**, University of Bedfordshire, UK
- **Eisenhardt, Monika**, Univeristy of Economics Katowice, Poland
- **Gabryelczyk, Renata**, University of Warsaw, Poland
- **Geri, Nitza**, The Open University of Israel, Israel
- **Leyh, Christian**, Technische Universität Dresden, Germany
- **Malgeri, Michele**, Universita' degli Studi di Catania, Italy
- **Muszyńska, Karolina**, University of Szczecin, Poland
- **Nikiforova, Anastasija**, University of Tartu, Estonia
- **Rizun, Nina**, Gdansk University of Technology, Poland
- **Rozevskis, Uldis**, University of Latvia, Latvia
- **Sobczak, Andrzej**, Warsaw School of Economics, Poland
- **Swacha, Jakub**, University of Szczecin, Poland
- **Symeonidis, Symeon**, Democritus Univesity of Thrace, Greece

- **Szczerbicki, Edward**, Newcastle University, Australia
- **Szumski, Oskar**, University of Warsaw, Faculty of Management, Poland
- **Wielki, Janusz**, Opole University of Technology, Poland
- **Wątróbski, Jarosław**, University of Szczecin, Poland
- **Zborowski, Marek**, University of Warsaw, Poland

Optimization of Processes for Shared Cars

Ivo Oditis, Viesturs Spulis, Zane Bicevska
[0000-0003-2354-3780, 0000-0003-3044-8335, 0000-0002-5252-7336]
DIVI Grupa Ltd
{Ivo.Oditis, Viesturs.Spulis, Zane.Bicevska}@di.lv

Janis Bicevskis
[0000-0001-5298-9859]
University of Latvia
Janis.Bicevskis@lu.lv

Abstract— This study is devoted to process optimization for commercial sharing of e-vehicles. The model describes a system with one-way trips and relocations of e-vehicles between sectors by service personnel according to a dynamically compiled list of service trips. The model includes an algorithm for increasing the expected income, depending on the dynamically selected e-vehicle transfer. The implementation of the MIP (Mixed-Integer Programming) type algorithm pays particular attention to its performance, as optimization should be performed dynamically within few hours' intervals. The developed model has been validated for its practical application in Riga, Latvia.
Keywords- Sharing of e-Vehicles, One-way Trips and Relocations, Mixed-Integer Programming.

I. INTRODUCTION

SUSTAINABLE development requirements in the EU provide guidelines for improving the transport system, first and foremost by reducing CO₂ emissions. One of the possible solutions to achieve the goal is to switch to the use of electric vehicles (e-vehicle). According to [1], e-vehicle sharing systems benefit both users and the society in general. The two main benefits for individual users include reduced personal transport costs and improved mobility. Research has shown that e-vehicle sharing reduces the average number of kilometers travelled by a vehicle and is likely to reduce traffic congestions [2] and CO₂ emissions [3].

Vehicle rental approaches tend to be classified in two large groups [4]: (1) traditional rental—when customers receive and transfer vehicles after use at specially arranged points of leasing firms and rental will take one or more days—and (2) vehicle sharing—when vehicles can be taken for use anywhere, even for a very short period of time, and may be left anywhere at the end of the trip. Vehicle sharing has quickly gained popularity. The growth rate of the service has increased during the COVID-19 pandemic, - as it allows urban populations to avoid the need to travel to their destination via public transport.

The main challenge facing e-vehicle sharing rental systems is to achieve the optimal (the most profitable) deployment of vehicles in a city. This requires relocating e-vehicles quickly to the most profitable sectors of the city which in

turn causes additional costs. The studies [5] show that technical relocation of vehicles take approximately 14% of the total distance carried out by the vehicles. Optimization algorithms are used to give recommendations to system holders on the need to relocate vehicles to achieve a “more cost-effective” deployment and, hence, higher returns.

This study offers an e-vehicle sharing model that considers the dynamics of relocating e-vehicles in a city. The proposed model is designed to fully meet the requirements of real systems and differs from all known solutions.

The paper is structured as follows: a theoretical background on vehicle sharing models (Section 2), an original vehicle sharing model proposed by the authors (Section 3), a short discussion on the research findings (Section 4), and conclusions (Section 5).

II. RELATED RESEARCH

A. Review of Sharing Models

Although scientific literature on e-vehicle sharing is broad, the authors of other works as well as the authors of this study conclude that the scientific literature currently available does not offer a model that, along with parameters such as the number, size and location of charging stations, the size of the car fleet, would also consider the dynamics of vehicle relocations and system balancing when reserving of e-vehicles is used. The existing models [6] and [7] either use station locations without considering vehicle relocations [7] or use station locations, assuming only a limited subset of stations corresponding to the current demand should be serviced [6]. If vehicle relocations are modeled vehicle movements and associated costs are only considered at the end of the operating period (usually daily) and, therefore, affect the size of the available fleet [1].

According to [8] which studies an example of a city in Southern California, even 3–6 vehicles can be sufficient to provide 100 trips daily and achieve optimal customer waiting times. Meanwhile, about 18–24 vehicles would be enough to reduce the number of vehicle relocations. The authors conclude that, in addition to the number of vehicles (per trip), the relocation algorithm and the charging approach used are key factors for the successful use of such a system.

Boyaci [1] highlights the importance of the service level, which, in his view, influences the access of potential users to

The research leading to these results has received funding from the research project "Competence Centre of Information and Communication Technologies" of EU Structural funds, contract No. 1.2.1.1/18/A/003 signed between IT Competence Centre and Central Finance and Contracting Agency, Research No. 1.15 "Use of artificial intelligence for optimization of e-mobility solutions".

e-vehicle stations, i.e. (1) the distance between the location of the e-vehicle and the destination, respectively, from the point of start and arrival of the e-vehicle, and (2) the availability of e-vehicles at stations. On the other hand, the number and size of the stations and the size and availability of the e-vehicle fleet at “real time” at the “particular station” are affected by the costs of establishing and operating the e-vehicle sharing system.

According to the classification of [4], the e-vehicle sharing system analyzed in this paper is:

- (1) a commercial solution as the aim is to generate maximum income,
- (2) station-based – e-vehicles are deployed in any available parking place and the city is divided into areas - stations,
- (3) one-directional as the customer is allowed to not return the e-vehicle to the start point of the trip,
- (4) with relocations as the service staff moves e-vehicles to potentially more favourable places in the city,
- (5) with dynamic booked trips as an e-vehicle may be rented by the customer anywhere, at any time without prior e-vehicle reservation.

Increased profits for commercial e-vehicle sharing can be achieved by supplying vehicles to the places in a city where customers will need them with the highest probability as well as increasing the relocation efficiency. The e-vehicle sharing models proposed by other authors differ significantly from those proposed in this work.

B. Optimization Algorithms

An optimization algorithm that was matching to the task of this study is the Mixed-Integer Programming (MIP) model, which maximizes profits on the assumption that next trips, the availability of fixed stations and availability of relocation staff are known.

The model described in [9] has a deficiency from model of this study. It is designed on the assumption that future trips are known, e.g. customers order the vehicles indicating starting and ending stations and the duration of the trip. Originally, the authors examined the possibility of predicting customer trips based on the data history of many previous trips. However, the experiments failed to obtain a sufficiently reliable forecast for future trips, so this idea was rejected. On the other hand, if the trips forecast is not sufficiently precise, the model defined by [9] does not provide a credible relocation plan, i.e., the vehicles will possibly be moved to places where customers will not need them.

Consequently, the [9] algorithm is not used directly in the study, and the authors have developed an original algorithm.

III. VEHICLE DEPLOYMENT MODEL

The model consists of several successive steps:

- dividing the city into sectors and estimating the costs of moving e-vehicles between sectors,
- identification of repeated trips,

- determining the value of an e-vehicle in a specific sector and time,
- forecasting of booked trips,
- an estimate of the total profitability of the e-vehicle at a given moment before relocation.
- compiling a list of profitable relocations,
- optimization of the relocation execution plan,
- estimation of the total profitability of the e-vehicle after relocation,
- creating of relocations plans.

In this chapter, the model will be discussed informal fleetly, leaving a description of the formalized model to other paper.

A. Station-based Algorithm

According to [9], the continuous division of the transport sharing service area into sectors (other studies referred to as stations) does not significantly affect optimization. Cluster analyzing the history of trips and knowing the specific characteristics of the area, Riga city was divided into sectors as can be seen in Fig. 1.



Fig. 1. Division of Riga, Latvia in sectors (using OpenStreetMap: www.openstreetmap.org/copyright)

However, division of territory into sectors must be carried out under several conditions. First, the sectors need to be relatively small to place a vehicle in the area for the client to reach it within “reasonable” time (the accumulated real data set shows that customers are ready to spend up to five minutes for reaching a vehicle). Secondly, the driving time between two adjacent sectors must be comparable. Thirdly, within one sector, customers' behavior must be comparable, i.e., customers make trips from the respective sector uniformly frequently.

B. Forecasting User Trips

While sharing e-vehicle users in Riga city do not make all requests a day in advance, certain user trips can be scheduled with high possibility, using historical trip data. Using cluster analysis, “routine arcs” can be found: regular trips

that consistently start and end the day from day to day in the same sectors, at approximately the same time.

C. Estimation of Sectors' Income

The purpose of relocation is to place e-vehicle in areas where they are in demand and profits are expected accordingly. In sharing systems with booking in advance, a full estimate of demand and expected profit is known prior to the planning of relocation operations. But in our case, requests are made in real time. Therefore, to take tactical decisions on relocation operations, it is necessary to be able to carry out an alternative assessment to which stations to move the e-vehicles.

One potential solution that this study looks at is the modelling of expected income using historical data. The model of expected incomes describes the average expected benefits over a specific time period from a e-vehicle parked in a particular sector that can be rented by users. The expected income most probably will vary from station to station, as well as it will change over a day: In "peak hours" the expected income will be higher, in "quiet hours" less.

Modelling the expected incomes may not only provide tactical support in the planning of resettlement operations, but also give general impression on the behavior of sharing e-vehicle users. Comparing the expected income at different times, different weekdays, and different stations, it will be possible to draw conclusions that can also help you to make strategic, long-term decisions, such as handling different sectors or deploying charging stations.

D. Estimation of Expected Income Using Historical Data

From the history of e-vehicle rentals in different sectors, you can do an assessment of expected income in each sector for the next day. This data is taken from historical information about e-vehicle rentals.

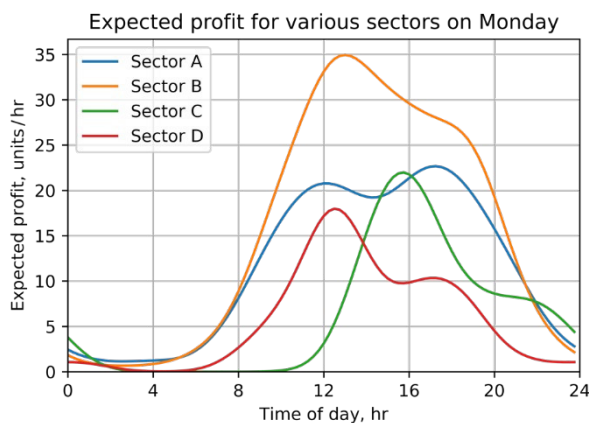


Fig. 2. Expected income for the sectors on Monday

Estimated forecasted incomes on Monday for the four sectors are summarized in Fig. 2. Data shows that there are significant differences between revenues for different sectors, as well as expected income changes by day. Moreover, some of the sectors can be very profitable only in specific time intervals and unprofitable in all others. For example, in the morning it is more convenient to move a car to sector D than

to sector C, but after 1 PM a car in Sector C will be more profitable than a car parked in Sector D.

E. Weekly variations

There is a difference in estimated income for a particular sector between weekdays. From simple assumptions about the behavior of e-vehicle sharing users, there can be expected that demand for shared cars, so expected income, could vary significantly between business days and holidays.

Indeed, such a phenomenon can be observed in the estimated expected income for the sector A, as shown in Fig. 3. Although there is a variation in the expected income between business days, there is a very significant difference in the expected income on holiday.

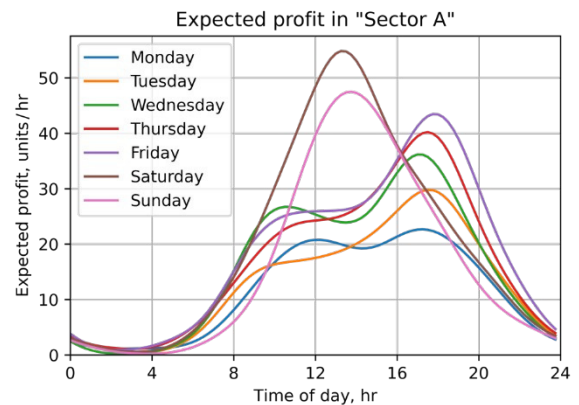


Fig. 3. Expected income in "Sector A" in different days of week

As there is a significant but hard-to-predict difference between weekdays, it is necessary to calculate the expected earnings for each day of the week separately.

F. Variation in Historical Data

A related subject matter related to the modelling of forecasted income from historical data is how old historical data is effectively used. Since the number of shared cars is limited, a full history may be considered for usage to reduce the "noise" in the data gathered. However, as a counter argument for the use of historical data, there can be mentioned that user demand for shared cars is not static but is changing in the result of seasonal or other long-term processes. It is also concluded that, to keep the calculation of the intended income used up to date, the expected in-come should be recalculated on a regular basis.

G. Limitations of the Proposed Method

While the method of estimating the expected income provides a valuable numerical estimate of the cost of parking sharing e-vehicle in specific sectors, effective use of the described method must be aware of its shortcomings and limitations. For example, the method may provide inaccurate results if too many e-vehicles are placed in a particular sector, or the area of the sector is too small. Similarly, it is necessary to have a reliable history of bookings to estimate the expected income, and it may not be available when starting a sharing e-vehicle operation in a new region.

The following chapters describe two main shortcomings and limitations for estimating of the expected income.

H. Linearity Assumption

In the definition and calculation process described above, there is an assumption that the estimated income for a particular e-vehicle parked in a sector does not depend on the total number of parked vehicles in that sector. At a large number of e-vehicle parked in the same sector, you can see that this assumption is flawed; if the number of e-vehicle parked in the sector significantly exceeds the demand for shared e-vehicle in this sector, the average per e-vehicle income will be low.

Information collected from historical data (see Fig. 4) allows you to analyze the veracity of the statement described. The graph shows the average number of rented cars in the sector F over two hours, depending on the number of parked cars in this sector.

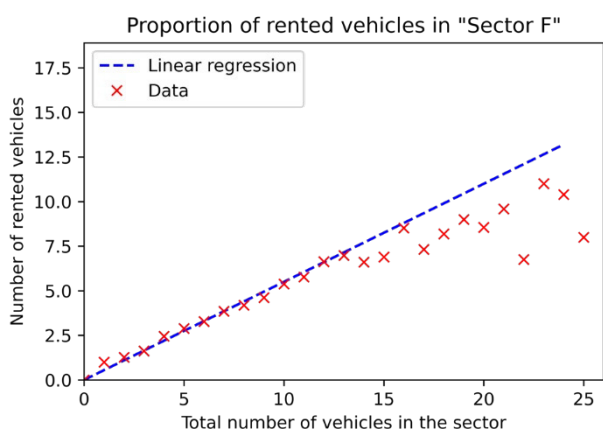


Fig. 4. Proportion of rented cars in "Sector F"

The picture shows that the percentage of cars rented in this sector is almost constant at a small number of cars (55% of the cars located in the sector will be rented within two hours). Hence, the possibility of renting a particular car does not depend on the total number of cars in the sector at low number of available cars. And therefore, the expected per car income does not depend on the total number of parked cars in the sector.

The breakpoint in the sector F, when the described revenue model is ineffective, is around 14 cars. However, the typical number of cars in the sector F is lower, so the described model for calculating the expected income is an acceptable approximation. Similar data analysis and finding a breakpoint can be used to find a flexible demand limit for each sector.

I. Usage of Historical Data

The described method for calculating of expected income is based on the existence of historical data. Unlike demographic-based models, the described model cannot be used when starting a sharing e-vehicle rental system in a new city or expanding operations into new sectors.

Reliance on the historical data prevents a model from rapid responding to changes in demand for shared e-vehicle,

or to price policies. The described expected income estimation algorithm will work most accurately if variations in the shared e-vehicle system are minimal.

Small sectors may lack data to adequately calculate expected income due to data noise. This phenomenon limits the lower size of sectors, thereby affecting the constructing of sectors.

J. Car-sharing Income Optimization by E-vehicle Relocation

Increasing income is possible by moving cars from low-income sectors to higher-potential sectors, but of course considering the costs of relocation. The location of e-vehicles at a given time determines the total value of all e-vehicles. This can be increased by moving the e-vehicle between sectors and finding the optimal location with the highest possible value.

The work of optimization is primarily inspired by the Mixed-Integer Programming (MIP) model proposed by C. Gambella, E. Malaguti, F. Masini, and D. Vigo for optimizing relocation operations in electric car-sharing [9]. The parameters discussed in the previous sections are passed to the optimization algorithm and it finds a new e-vehicle location by sectors that give the highest potential revenue, as well as a relocation plan to obtain this location.

In the case of many sectors and e-vehicles, the work of the algorithm may require significant computational resources/time. Therefore, it was assumed that the optimization algorithm was given a time limit during which the best of the considered variants is found, without guaranteeing to find the optimal solution.

As a result of the optimization algorithm execution, three reports are generated:

- a plan for relocation of e-vehicles between sectors,
- work plan for e-vehicle relocators,
- potential income change report.

IV. RESEARCH FINDINGS

The proposed solution uses historical data of shared e-vehicles — the intensity of trips across different sectors of the city, depending on season, day of week and clock time. This data can be obtained by recording events that the service provider information system can manage. The division of the city into sectors is also carried out using historical data, which in turn affects the permissible set of relocations and their costs. The model described is therefore applicable after the introduction of a shared transport service and the accumulation of historical data.

It should be acknowledged that e-vehicle rental services can be organized in many ways. The company offering services in Riga city, Latvia, provides a dynamic optimization of e-vehicle relocation approach. Obviously, the service capabilities determine the complexity of the model and its effectiveness. The rapid development of IoT will offer ever-new service capabilities that will require ever-new solutions.

This calls into question the need for a single, universal solution.

To estimate the potential income growth that can be achieved by moving e-vehicles between sectors, the relocation for a different number of e-vehicles and sectors was simulated (see Table 1). In the result of the simulation, there was concluded that e-vehicle relocation increases the total income by more than 15%.

Table 1. Potential income increase (results of simulation)

Number of vehicles	Number of relocators	Number of sectors	Number of relocation	Potential income	
				Before	After
12	2	30	9	82.44	167.03
40	5	30	9	502.20	571.85
56	7	30	23	552.72	583.50
60	5	57	22	781.56	952.10

The potential income calculated in the table is an estimate of all cars in the sectors combined. The potential income is, of course, indicative and cannot be determined as an absolute value but its changes show the relative effect of relocations on the potential income.

V. CONCLUSIONS

The study offers a model for the use of shared e-vehicles, described as a commercial system with one-way trips and dynamic relocations of e-vehicles between city sectors, without pre-booked trips. The model consists of the following set of parameters: breakdown of the city in sectors, maximal number of available cars, number of cars per sector, set of possible relocations, parameters characterizing e-vehicle relocations, number of available relocators.

The parameters used in the model allow to describe the operation of a real system:

- The strategic level determines the number of e-vehicle available for sharing and the maximum number of e-vehicle to be placed in each urban sector.
- At the tactical level, historical shared vehicle data allows calculating the profitability of e-vehicle depending on the season, day of the week, usage time and city sector in which e-vehicle is placed.
- At the operational level, the total daily income is estimated as a sum of the average expected income over a specific period of time from all e-vehicle placed in a specific sector that can be rented by users.

The study provides an algorithm that optimizes expected income based on the set of selected relocations using the values of the above parameters. When implementing an algorithm, special attention should be paid to its performance as

optimization must be performed dynamically, within few hours' interval.

The vehicle sharing model proposed in the study is only one step towards an optimal solution. The model only partly describes real-life processes, such as e-vehicle battery capacity and technical parameters and prices. Similarly, the model does not consider cases where it is beneficial for several customers to use the same e-vehicle when a route is agreed. In addition, the work does not analyze the risks posed by concurrent usage of shared e-vehicle. Such a study has been conducted for e-commerce [10] and e-scooters [11]. These issues may be the content of further studies.

REFERENCES

- [1] H. Boyacı, K. G. Zografos, and N. Geroliminis, "An optimization framework for the development of efficient one-way car-sharing systems," in *European Journal of Operational Research*, vol. 240(3), pp. 718-733, 2015. <https://doi.org/10.1016/j.ejor.2014.07.020>
- [2] K. Crane, L. Ecola, S. Hassell, and S. Natarah, "An alternative approach for identifying opportunities to reduce emissions of greenhouse gases," *Tech. rep.*, RAND Corporation, 2012. https://www.rand.org/pubs/technical_reports/TR1170.html
- [3] S. A. Shaheen, and A. P. Cohen, "Carsharing and personal vehicle services: worldwide market developments and emerging trends." *International journal of sustainable transportation*, vol. 7(1), pp. 5-34, 2013, DOI: 10.1080/15568318.2012.660103
- [4] S. Illgen, and M. Höck, "Literature review of the vehicle relocation problem in one-way car sharing networks," *Transportation Research Part B: Methodological*, vol. 120, 2019, pp. 193-204. <https://doi.org/10.1016/j.trb.2018.12.006>
- [5] A. S. Vasconcelos, L. M. Martinez, G. H. Correia, D. C. Guimarães, and T. L. Farias, "Environmental and financial impacts of adopting alternative vehicle technologies and relocation strategies in station-based one-way carsharing: An application in the city of Lisbon, Portugal," *Transportation Research Part D: Transport and Environment*, vol. 57, 2017, pp. 350-362. <https://doi.org/10.1016/j.trd.2017.08.019>
- [6] G. H. de Almeida Correia, and A. P. Antunes, "Optimization approach to depot location and trip selection in one-way carsharing systems," *Transportation Research Part E: Logistics and Transportation Review*, vol. 48(1), 2012, pp. 233-247.
- [7] J. R. Lin, and T. H. Yang, "Strategic design of public bicycle sharing systems with service level constraints." *Transportation research part E: logistics and transportation review*, vol. 47(2), 2011, pp. 284-294. <https://doi.org/10.1016/j.tre.2010.09.004>
- [8] G. Brandstätter, C. Gambella, M. Leitner, E. Malaguti, F. Masini, J. Puchinger, and D. Vigo, "Overview of optimization problems in electric car-sharing system design and management. In *Dynamic perspectives on managerial decision making*," Springer Cham., pp. 441-471, 2016. DOI: 10.1007/978-3-319-39120-5_24
- [9] C. Gambella, E. Malaguti, F. Masini, and D. Vigo, "Optimizing relocation operations in electric car-sharing," *Omega*, vol. 81, 2018, pp. 234-245. <https://doi.org/10.1016/j.omega.2017.11.007>
- [10] J. Bicevskis, A. Nikiforova, G. Karnitis, I. Oditis, and Z. Bicevska, "Risks of Concurrent Execution in E-Commerce Processes," In *Proceedings of the 16th Conference on Computer Science and Intelligence Systems. ACSIS, Vol. 25*, pp. 447-451, 2021.
- [11] J. Bicevskis, G. Karnitis, Z. Bicevska, and I. Oditis, "Analysis of Concurrent Processes in Internet of Things Solutions," *Information Technology for Management: Business and Social Issues: 16th Conference, ISM 2021 and FedCSIS-AIST 2021 Track, Held as Part of FedCSIS 2021, September 2-5, 2021: Extended and Revised Selected Papers / eds.: Ewa Ziemia, Witold Chmielarz. LNBI*, vol. 442. Cham: Springer, 2022, pp. 26-41. https://doi.org/10.1007/978-3-030-98997-2_2

Towards Temporal Multi-Criteria Assessment of Sustainable RES Exploitation in European Countries

Aleksandra Bączkiewicz

Institute of Management, University of Szczecin
ul. Cukrowa 8, 71-004 Szczecin, Poland
Email: aleksandra.baczkiewicz@phd.usz.edu.pl

Abstract—This paper aims to introduce a novel Temporal SWARA-SPOTIS method for multi-criteria temporal assessment. The proposed method combines the Step-Wise Weights Assessment Ratio Analysis (SWARA) method for determining the significance values of particular periods and the Stable Preference Ordering Towards Ideal Solution (SPOTIS) method for the multi-criteria assessment. The developed method was applied for assessing the sustainable use of renewable energy sources (RES) by European countries in various branches of the economy and industry, considering multiple criteria and the dynamics of results change over time. The application of the proposed method is presented in an illustrative example covering the assessment of 30 selected European countries over the five years 2015-2019. The presented approach proved its usefulness in the problem investigated and provided reliable results indicating that the best-scored countries regarding sustainable use of RES are dominantly the Nordic countries.

I. INTRODUCTION

RENEWABLE energy sources (RES) play an essential role in the sustainable economy. The increase in RES participation in various domains contributes to limiting greenhouse gas and pollutants emissions and reducing countries' dependence on imports of non-renewable energy sources. The efforts to increase the RES share cover different dimensions. Among them is electricity generation from RES such as Hydro, Wind, Solar, Biomass, Geothermal, and Wave (tidal). Besides, energy policies promoting RES usage include increasing the RES share in energy consumption in transportation and heating and cooling sectors. Thus, appropriate measurement tools are necessary to assess the achievement of planned goals and evaluate regions [1].

Reliable assessment of sustainable RES use requires simultaneous consideration of dimensions such as economic, environmental, and social [1]. The assessment methodology for multi-criteria RES problems should consider different aspects, such as various types of RES, several attributes of the location for RES-generating infrastructures, and different sectors in which RES are produced and consumed [2]. Multi-criteria decision analysis methods (MCDA) fulfill these requirements [3]. Many research papers are focused on evaluating RES problems. Multi-criteria assessment of countries in terms of preparation for the sustainable energy transition was performed using Preference Ranking Organization Method for Enrichment of Evaluation (PROMETHEE) II and Analytical Hierarchy Process (AHP) [1]. A comparative analysis

employing Characteristic Objects Method (COMET), Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS), Vise Kriterijumska Optimizacija I Kompromisno Resenje (VIKOR), and PROMETHEE II, was conducted to assess the European countries in terms of energy consumption with particular attention to RES share [3]. MCDA methods were applied to evaluate infrastructure and technologies for generating electricity from RES. COMET and Stable Preference Ordering Towards Ideal Solution (SPOTIS) were used to evaluate solar panel alternatives regarding selected technical attributes of assessed options [4].

The literature review confirms the usefulness of MCDA methods in the multi-dimensional evaluation of RES for a single moment. However, a clear research gap is visible, including the lack of simultaneous respect for the performance variability over the time analyzed. Several MCDA attempts of temporal approach can be found in the literature, like the TOPSIS-based approach considering the variability of results over time. This approach considers evaluating alternatives using TOPSIS individually for each analyzed year. Results are re-evaluated using TOPSIS and weights assigned to years [5]. The authors of another research adapted the PROMETHEE II method to perform a multi-criteria evaluation of temporal sustainable forest management [6]. This approach aggregates the results of comparing pairs of criteria in each period. The rank relations are converted to preference relations for each pair of alternatives and each period in the next stage. However, the procedure described is complex, making applying to complex hierarchical models containing multiple criteria challenging. This paper introduces the Temporal SWARA-SPOTIS method developed for multi-criteria temporal evaluation. The application of the proposed method is illustrated in the example of a temporal multi-criteria assessment of selected European countries in terms of RES use in various branches of the economy and industry.

The rest of the paper is organized as follows. Section II gives the background and formulas for SWARA-SPOTIS. The following section III introduces the practical problem of sustainability assessment focused on RES exploitation by European countries is introduced. The next section IV presents and discusses research results. Finally, in the last section V conclusions are provided, and future work directions are drawn.

II. METHODOLOGY

A. The Temporal SWARA-SPOTIS method

Step 1. Create the temporal decision matrix $S = [s_{ip}]_{m \times t}$ including in columns the utility function values (weighted normalized average distance values) calculated by SPOTIS for each i th alternative $i = 1, 2, \dots, m$ in p th periods analyzed, where $p = 1, 2, \dots, t$. The SPOTIS steps are presented in [7]. In this research criteria weights were determined using objective weighting method called CRITIC demonstrated in [8].

Step 2. This step involves determining the significance of particular periods using SWARA [9]. Rank periods in descending order according to their significance. Period p_1 is the most significant.

Step 3. Establish comparative importance ratio c among investigated periods. Start with the period p_2 and define how much period p_1 is more significant than p_2 . Determine c_p ratio using values in the range from 0 to 1, analogously to percentage. Value of comparative importance ratio c_1 is determined for periods p_1 and p_2 . Then, identical procedure is followed up to period p_t . Comparative importance determined between p_{t-1} and p_t is denoted by c_{t-1} , where t represents number of all periods to investigate.

Step 4. Compute the coefficient k_p values according to Equation (1), where p represents periods ranked in descending order according to their importance.

$$k_p = \begin{cases} 1, & p = 1 \\ c_p + 1, & p > 1 \end{cases} \quad (1)$$

Step 5. Calculate initial weights v_p for particular periods as Equation (2) presents.

$$v_p = \begin{cases} 1, & p = 1 \\ \frac{v_{p-1}}{k_p}, & p > 1 \end{cases} \quad (2)$$

Step 6. Determine final SWARA weights w_p for each period according to Equation (3).

$$w_p = \frac{v_p}{\sum_{p=1}^t v_p} \quad (3)$$

Step 7. The three final stages involve the Temporal SWARA-SPOTIS assessment of matrix S including SPOTIS utility function values s in the form of weighted average distance values calculated for alternatives for each period p . First step includes determination of the normalized distances d_{ip} for each alternative A_i from Ideal Solution Point S^* according to Equation 4. S^* is represented by S^{min} since the SPOTIS creates rankings by sorting alternatives in ascending order, considering utility function values received by alternatives in each period. Alternative with the lowest utility function value is regarded as the best-evaluated option.

$$d_{ip}(A_i, s_p^*) = \frac{|s_{ip} - s_p^*|}{|s_p^{max} - s_p^{min}|} \quad (4)$$

Step 8. Compute the final temporal utility function values for each alternative as Equation (5) shows

$$d(A_i, s^*) = \sum_{p=1}^t w_p d_{ip}(A_i, s_p^*) \quad (5)$$

where w_p represents SWARA weights assigned for particular periods.

Step 9. Generate the final Temporal SWARA-SPOTIS ranking of evaluated alternatives involving the full investigated time by sorting values $d(A_i, s^*)$ obtained in the previous step in increasing order. The best-evaluated option has the lowest $d(A_i, s^*)$ value. Rankings are compared using two correlation coefficients: Weighted Spearman rank correlation coefficient r_w described in [4] and Spearman rank correlation coefficient detailed in [10]

III. THE PRACTICAL PROBLEM OF EUROPEAN COUNTRIES' CONSIDERING TEMPORAL ASSESSMENT OF RES USAGE

The framework for temporal assessment of sustainable RES using is based on annual data provided by Eurostat in a database collected with the SHARES (SHort Assessment of Renewable Energy Sources) tool [11]. Particular criteria are included in Table I.

TABLE I
CRITERIA FOR SUSTAINABLE RES USING ASSESSMENT.

C_j	Criterion name	Goal	Unit
C_1	Annual electricity generation from Hydro	Max	[% of E]
C_2	Annual electricity generation from Wind	Max	[% of E]
C_3	Annual electricity generation from Solar	Max	[% of E]
C_4	Annual electricity generation from Solid biofuels	Max	[% of E]
C_5	Annual electricity generation from all other renewables	Max	[% of E]
C_6	Annual consumption of renewable electricity in road transport	Max	[% of T]
C_7	Annual consumption of renewable electricity in rail transport	Max	[% of T]
C_8	Annual consumption of renewable electricity in all other transport modes	Max	[% of T]
C_9	Annual consumption of renewable electricity from compliant biofuels in transport	Max	[% of T]
C_{10}	Annual final energy consumption in heating and cooling	Max	[% of H&C]
C_{11}	Annual derived RES based heat in heating and cooling	Max	[% of H&C]
C_{12}	Annual derived RES based heat in heating and cooling for heat pumps	Max	[% of H&C]
C_{13}	Gross final consumption of energy from renewable sources in electricity	Max	[% of G]
C_{14}	Gross final consumption of energy from renewable sources in heating and cooling	Max	[% of G]
C_{15}	Gross final consumption of energy from renewable sources in transport	Max	[% of G]

There are criteria covering generation of electricity from RES (C_1 – C_5) and its consumption (C_6 – C_{15}). Data in the mentioned database are available in the unit KTOE (Thousand tonnes of oil equivalent). However, in an attempt to provide a more reliable and objective assessment, this research employed percentage data representing the share of each measure in each sector. This approach enables the reduction of inequalities between the countries caused by non-modifiable factors such

as area, geographical location, and population, which objectifies the assessment. Therefore, this framework considers RES percentage share in sectors considering all energy sources, such as electricity production (E), energy consumption in transport (T), heating and cooling (H&C), and gross final energy consumption (G). The goal of each criterion is maximization because the assumption of sustainable development is to increase the share of RES in all sectors.

Performance values in the form of percentages of criteria representing the use of RES in particular sectors for 2015–2019 are available in the GitHub repository at [12] in a dataset folder. The results of a multi-criteria temporal assessment concerning sustainable RES are presented in the following section IV.

IV. RESULTS

This section presents the results of the temporal multi-criteria assessment of RES exploitation in European countries performed by the SWARA-SPOTIS method. Criteria weights were determined for each year using the CRITIC method. Then, each decision matrix was evaluated by the SPOTIS method. Next, a decision matrix containing utility function values obtained by countries in each year was created. The next step was determining the significance values for each period using the SWARA method. Then, a decision matrix including SPOTIS utility function values for each year was evaluated using the SWARA weights. The resulting vector with Temporal SWARA-SPOTIS utility function values for each country aggregates annual results into a single score. The obtained Temporal SWARA-SPOTIS utility function values were then ranked in ascending order, according to the SPOTIS rule. It can be observed that Sweden (A_{27}) is the leader of both rankings in all years analyzed. Thus, Sweden is expected to be the ranking leader aggregating the grades achieved in the analyzed period. For the other countries, performing a reliable assessment incorporating the dynamics of performance changes over time is no longer straightforward and intuitive. Instead, it requires using an appropriate methodology, such as Temporal SWARA-SPOTIS. Table II contains the results of the subsequent stages of the SWARA method applied to determine the significance of particular periods.

TABLE II
SWARA WEIGHTS OF PARTICULAR YEARS INVESTIGATED.

Year	c_p	k_p	v_p	w_p
2019	-	1	1.0000	0.3839
2018	0.5	1.5	0.6667	0.2559
2017	0.5	1.5	0.4444	0.1706
2016	0.5	1.5	0.2963	0.1137
2015	0.5	1.5	0.1975	0.0758

The most recent year is considered the most significant, while for the earlier years, the significance gradually decreases. In applied strategy, each subsequent year is 50% more significant than the year preceding. The advantage of the proposed method is that the decision-maker can arbitrarily

model the relevance of each period by setting values of comparative importance ratio c_p for each period. It implies that 2016 is 50% more significant than 2015. For subsequent years, the procedure is analogous. Column w_p contains the final SWARA weights calculated for each period p . Table III includes annual SPOTIS rankings calculated for each country in each period. Columns "TSS" contain rankings provided by the Temporal SWARA-SPOTIS method. Scores (utility function values) of SPOTIS are provided on [12] in folder called results.

TABLE III
RESULTS OF CLASSICAL SPOTIS AND TEMPORAL SWARA-SPOTIS FOR 2015–2019.

A_i	Country	2015	2016	2017	2018	2019	TSS
A_1	Belgium	24	22	23	24	24	24
A_2	Bulgaria	12	14	16	14	15	14
A_3	Czechia	16	18	20	20	20	19
A_4	Denmark	6	4	3	2	2	3
A_5	Germany	10	10	11	11	11	11
A_6	Estonia	13	11	12	10	10	10
A_7	Greece	14	15	10	12	12	12
A_8	Spain	17	13	13	15	14	13
A_9	France	18	19	19	21	19	20
A_{10}	Croatia	19	17	18	18	17	17
A_{11}	Ireland	29	29	28	30	30	30
A_{12}	Italy	7	7	7	9	8	7
A_{13}	Cyprus	27	26	24	17	21	23
A_{14}	Latvia	9	9	9	8	9	9
A_{15}	Lithuania	15	16	17	19	22	18
A_{16}	Luxembourg	28	28	30	29	29	29
A_{17}	Hungary	22	24	25	25	26	26
A_{18}	Malta	23	20	15	13	13	15
A_{19}	Netherlands	30	30	29	27	27	27
A_{20}	Austria	2	2	4	3	3	2
A_{21}	Poland	25	27	27	28	28	28
A_{22}	Portugal	5	5	6	6	6	6
A_{23}	Romania	11	12	14	16	16	16
A_{24}	Slovenia	20	21	21	22	23	21
A_{25}	Slovakia	21	25	26	26	25	25
A_{26}	Finland	3	6	5	5	5	5
A_{27}	Sweden	1	1	1	1	1	1
A_{28}	United Kingdom	26	23	22	23	18	22
A_{29}	Iceland	8	8	8	7	7	8
A_{30}	Norway	4	3	2	4	4	4

As expected, Sweden (A_{27}) is the best-scored country regarding the sustainable share and use of RES. Austria (A_{20}) took second place. Austria ranked second in 2015 and 2016, dropped to fourth in 2017, and ranked third in 2018 and 2019, despite the worsening performance in 2017–2019. However, the Temporal SWARA-SPOTIS method employs the utility function values obtained in the individual years as performance values, which are more precise than ranks. This feature allows for a more accurate and reliable reflection of the aggregate performance of the countries over the years reviewed. Denmark achieved third place (A_4). This country improved the use of RES in the economy over the years analyzed. It ranked sixth in SPOTIS in 2015, then jumped to fourth place in 2016. In 2017, there was a further promotion of Denmark to third place. In 2018, Denmark again climbed to second place and remained there in 2019. Because most recent years are more relevant, the promotions registered between

2017 and 2019 allowed Denmark to reach the third position in the final ranking despite the sixth place occupied in 2015. Norway (A_{30}) took fourth place in the final ranking. Norway in 2015 was fourth. In 2016, Norway moved up to third place and in 2017 to second place. However, it was again ranked fourth in 2018 and 2019. The greater importance of most recent years caused the better performance in 2016-2017 did not enable Norway to rank higher than fourth in the final ranking. Finland (A_{26}) received fifth place in the final ranking. This country was ranked third in 2015. Then in 2016, Finland dropped to sixth place. In contrast, Norway advanced to fifth place in 2017. Therefore, this country retained a fifth place in the remaining years analyzed. Table IV contains the values of the correlation coefficients r_w and r_s representing the convergence of the final aggregated rankings obtained using Temporal SWARA-SPOTIS with the SPOTIS rankings generated for the individual years analyzed. High values of both correlation coefficients close to 1 indicate high convergence of the compared rankings.

TABLE IV
CORRELATION OF TEMPORAL SWARA-SPOTIS WITH SPOTIS.

Year	2015	2016	2017	2018	2019
r_w	0.9562	0.9835	0.9912	0.9885	0.9906
r_s	0.9448	0.9795	0.9907	0.9867	0.9884

The results confirm that the Temporal SWARA-SPOTIS ranking is more convergent with the most recent analyzed years, 2017-2019, than with the earlier years, 2015-2016. Results are consistent with the assumption that the most recent years are more interesting for decision-makers and reflect appropriately the influence of the weights assigned to the following years by the SWARA method.

V. CONCLUSIONS

This paper demonstrated the application of the newly developed Temporal SWARA-SPOTIS method on the illustrative example of a multi-criteria problem involving evaluating the sustainable use of RES by European countries, considering the dynamics of performance variability over the observed five years. The developed methodology indicated Sweden as the most sustainable country among the investigated European countries. Likewise, other Nordic countries such as Denmark, Norway, and Finland are among the best-ranked countries. Austria is also a well-scored country. The proposed tool has a high potential of usefulness for information systems supporting multi-criteria sustainability assessment taking into account both multiple indicators and dimensions and the variability of results over time.

The proven usefulness of the proposed tool suggests extending the conducted research to explore other MCDA methods and techniques for determining the relevance of periods. An interesting future work direction seems to be an approach adapting the PROMETHEE II method for multi-criteria temporal sustainability assessment. This method appears promising due to its ability to employ different preference functions

and limited criteria compensation. Further research focused on temporal multi-criteria sustainability assessment is also planned to include a study of the impact of other objective criteria weighting methods on the results. Investigating the utility of the proposed sustainability assessment approach based on other RES indicators may also be an interesting research direction.

ACKNOWLEDGMENT

The work was supported by the project financed within the framework of the program of the Minister of Science and Higher Education under the name "Regional Excellence Initiative" in the years 2019-2022, Project Number 001/RID/2018/19; the amount of financing: PLN 10.684.000,00.

REFERENCES

- [1] H. Neofytou, A. Nikas, and H. Doukas, "Sustainable energy transition readiness: A multicriteria assessment index," *Renewable and Sustainable Energy Reviews*, vol. 131, p. 109988, 2020. doi: <https://doi.org/10.1016/j.rser.2020.109988>
- [2] Z. Andreopoulou, C. Koliouka, E. Galarotis, and C. Zopounidis, "Renewable energy sources: Using PROMETHEE II for ranking websites to support market opportunities," *Technological Forecasting and Social Change*, vol. 131, pp. 31-37, 2018. doi: <https://doi.org/10.1016/j.rser.2017.05.190>
- [3] A. Bączkiewicz and B. Kizielewicz, "Towards Sustainable Energy Consumption Evaluation in Europe for Industrial Sector Based on MCDA Methods," *Procedia Computer Science*, vol. 192, pp. 1334-1346, 2021. doi: <https://doi.org/10.1016/j.procs.2021.08.137>
- [4] A. Bączkiewicz, B. Kizielewicz, A. Shekhovtsov, M. Yelmikheiev, V. Kozlov, and W. Sałabun, "Comparative analysis of solar panels with determination of local significance levels of criteria using the MCDM methods resistant to the rank reversal phenomenon," *Energies*, vol. 14, no. 18, p. 5727, 2021. doi: <https://doi.org/10.3390/en14185727>
- [5] A. Frini and S. Benamor, "Making decisions in a sustainable development context: A state-of-the-art survey and proposal of a multi-period single synthesizing criterion approach," *Computational Economics*, vol. 52, no. 2, pp. 341-385, 2018. doi: <https://doi.org/10.1007/s10614-017-9677-5>
- [6] B. Urli, A. Frini, and S. B. Amor, "PROMETHEE-MP: a generalisation of PROMETHEE for multi-period evaluations under uncertainty," *International Journal of Multicriteria Decision Making*, vol. 8, no. 1, pp. 13-37, 2019. doi: <https://dx.doi.org/10.1504/IJCDM.2019.098042>
- [7] J. Dezert, A. Tchamova, D. Han, and J.-M. Tacnet, "The spotis rank reversal free method for multi-criteria decision-making support," in *2020 IEEE 23rd International Conference on Information Fusion (FUSION)*. IEEE, 2020. doi: <https://doi.org/10.23919/FUSION45008.2020.9190347> pp. 1-8.
- [8] A. Tuş and E. A. Adalı, "The new combination with CRITIC and WASPAS methods for the time and attendance software selection problem," *Opsearch*, vol. 56, no. 2, pp. 528-538, 2019. doi: <https://doi.org/10.1007/s12597-019-00371-6>
- [9] S. H. Zolfani and P. Chatterjee, "Comparative evaluation of sustainable design based on Step-Wise Weight Assessment Ratio Analysis (SWARA) and Best Worst Method (BWM) methods: a perspective on household furnishing materials," *Symmetry*, vol. 11, no. 1, p. 74, 2019. doi: <https://doi.org/10.3390/sym11010074>
- [10] M. Sajjad, W. Sałabun, S. Faizi, M. Ismail, and J. Wątróbski, "Statistical and analytical approach of multi-criteria group decision-making based on the correlation coefficient under intuitionistic 2-tuple fuzzy linguistic environment," *Expert Systems with Applications*, vol. 193, p. 116341, 2022. doi: <https://doi.org/10.1016/j.eswa.2021.116341>
- [11] Eurostat, *Energy from renewable sources*, 2022. [Online]. Available: <https://ec.europa.eu/eurostat/web/energy/data/shares>
- [12] energyinpython, *Towards the Temporal Multi-Criteria Assessment of Sustainable RES Exploitation in European Countries*, 2022. [Online]. Available: <https://github.com/energyinpython/fedcsis-2022-RES>

Process-oriented documentation of user requirements for analytical applications—challenges, state of the art and evaluation of a service-based configuration approach

Christian Hrach
 Institute for Applied
 Informatics, Goedelerring 9,
 04109 Leipzig, Germany,
 Email: hrach@infai.org

Rainer Alt
 Leipzig University,
 Grimmaische Str. 12, 04109
 Leipzig, Germany, Email:
 rainer.alt@uni-leipzig.de

Stefan Sackmann
 Martin Luther University Halle-Wittenberg,
 Universitätsring 3, 06108 Halle, Germany,
 Email: stefan.sackmann@wiwi.uni-halle.de

Abstract—In recent years, the integration of process design in conjunction with the use of analytical applications to provide information tailored to user requirements to support operational process activities (e.g., Operational BI) has become increasingly widespread. In analytical software development/implementation projects, the insufficient involvement of analytical end users with their process context and the resulting unclear requirements/expected analytical software functions are still one of the main reasons for analytical project failure. Embedded in a Design Science Research Process, this paper shows the shortcomings of existing approaches, tools and models (1. BPMN process model extensions, 2. configurators in analytical applications, 3. models used in analytical implementation projects) for the documentation/conceptual configuration of analytical requirements. As a second part, this paper presents the evaluation results of a new process-oriented and service-based configuration approach for analytical applications, whose practicability, usefulness and acceptance were evaluated in expert reviews and in analytical development projects.

I. INTRODUCTION

A. Main topic and challenges

PROCESS orientation has established itself in corporate practice since the 1990s as a primary structuring approach for corporate organizational forms and as the basis for a (re)organization of operational value-adding activities. In addition to the efficient linking of tasks, which is the initial focus of process-oriented design, the targeted use and visualization of information required in processes is becoming increasingly important [1] in times of advancing digitization and automation of corporate processes [2].

Systems of insight [3] or rather analytical applications have long been used for retrospective analysis of corporate activities for the management under the keyword Business Intelligence (BI). Due to a wider dissemination of analytical information in operational processes and the subsequent process-centric design of static analytical reports and interactive dashboards [4], original focus and functional range of analytical application/BI design [5] have widened. Additional aspects include 1. the trend to a modular analytical application structure [6], 2. the consideration of analytical self-

services [7] and 3. the implementation of real-time monitoring and automatic actions [8], [2] (Section III.A).

Concerning the development of analytical applications, the insufficient inclusion of end users with their process context and the resulting unclear requirements/expected deliverables are repeatedly cited in literature as the main causes for project failure or for the implementation of analytical applications that does not meet user expectations [9], [10]. Own results from interviews and an online survey with experts in analytical requirements management confirm this hypothesis: five of the experts surveyed find inadequately communicated/documented requirements are very frequent, four experts find them rather frequent and only two experts find them less frequent as a main reason for delayed or insufficient implementations of analytical applications.

In addition, practice-oriented literature provides evidence for a number of detailed causes for delays or failure of analytical development projects. Some of them already point out important aspects that must be better taken into account in future user- and process-oriented conceptual analytical application configuration in terms of requirements documentation. These include:

1. Unclear ideas on the users' side about goals, functionalities and detailed system specifications due to insufficiently elaborated and planned project scopes, information needs and use cases [11], [10];
2. Requirements formulated by users in an unclear and/or misleading manner and the resulting misunderstandings among technical developers [12], [9];
3. Data privacy risks known to the process users and not considered right from the start throughout requirements elicitation and documentation [9], [13];
4. Uncoordinated planning and implementation of individual data analyses within overall processes [14] leading to fragmented and insufficiently integrated analytical application landscapes and impeding the data-driven process design.

These aspects suggest the fact that models, configurators and other tools used in the documentation of requirements for process-related analytical applications in implementation projects are apparently insufficient.

B. Objective of the current work

In the context of the above-mentioned challenges, the results presented in this paper show a broad overview about the current state of the art concerning tools and models for configuration and conceptual modeling of analytical applications. The new results extend and supplement an earlier literature review regarding scientific models supporting requirements documentation for analytical applications (c.f. [15], summary in Section III.B) and investigate additional alternatives to configure analytical applications:

- Applicability of process modeling languages to represent information requirements (Section III.C);
- Availability of models for requirements documentation/conceptual configuration provided with analytical application products (Section III.D);
- Use of tools for requirements documentation in analytical development projects (Section III.E).

The results confirm the need for new tools to document/configure user requirements regarding analytical applications from a process-oriented perspective. As a second part of this paper, the authors present detailed results of expert reviews regarding the completeness and usefulness of a new approach for the conceptual configuration of analytical applications based on analytical services (initially presented in [15], short introduction in Section IV.A) and its practicability in the context of process-related analytical requirements documentation proofed in two real projects. This leads to the following two research questions:

RQ1: What support do models and tools from science and practice provide regarding documentation/configuration of process user requirements for analytical applications?

RQ2: To what extent is a configuration approach for analytical services suitable to be used in analytical implementation projects and to solve the identified challenges regarding documentation/configuration of process user requirements for analytical applications?

C. Structure of this paper

After the introduction of the subject area of this paper, the identification of current challenges and relevant process-oriented trends as well as the presentation of the paper's objectives in Section I, Section II presents the research method. Section III provides the current state of the art regarding different tools and models for documenting conceptual requirements for analytical process support in science and practice. Section IV shows the practical relevance of the new process-oriented configuration approach for analytical applications (initially introduced in [15]) based on evaluation results. Section V concludes the paper.

II. RESEARCH METHOD

The research results presented in this article have been elaborated as parts of a wider research project developing a

comprehensive modeling approach for the documentation/configuration of conceptual process user requirements for analytical applications (for further details see [15]). A six-step Design Science Research Methodology Process [16] guides this superordinate research project. Within the first Design Science phase "Problem Identification and Motivation", the following research methods were used to develop the new research results presented in the current paper:

- The modeling language Business Process Model and Notation (BPMN) served as a starting point to find suitable representations to configure analytical process requirements due to its leading position as a worldwide used process-modeling standard [17], [18]. The analysis (Section III.C) encompassed all previous BPMN extensions surveyed by [19].
- Analytical products were analyzed to determine whether they provide pre-built models/configurators for documenting business user requirements (Section III.D). This analysis comprised all 13 analytical products in the quadrants "Leaders", "Visionaries" or "Challengers" of "Gartner Magic Quadrant for Analytics and BI Platforms 2021" [20]. Investigation techniques included both interviews with software providers and the analysis of product information.
- Four interviews were conducted with project managers/experts responsible for requirements management. They confirmed that requirements documentation is still a critical, and so far an insufficiently supported success factor in analytical implementation projects. In addition, an online survey conducted with six experts in analytical requirements management from practice (about 125 participants of an event for analytics specialists were invited via e-mail) provided further feedback regarding the quality of different models, notations and documents used in this area (Section III.E).

In the fifth phase "Evaluation" of the overall Design Science Research Process, the previously developed configuration approach for analytic services and its intended integration into process models [15] was evaluated with regard to its usefulness/practical value (proof of value) [21] and its acceptance by the model users (proof of acceptance) [21], [22]. For this purpose, content-related feedback on the model structure (i.e., regarding the selected analytical services, their relationships to each other (service network) and the design of the service-internal service features) was collected in 12 expert reviews [21] with 17 experts in the field of analytics (project/requirements managers, data scientists). After the final definition of the analytical service network structure (after the third expert review), 13 experts provided an additional quality assessment (Section IV.B). Furthermore, the configuration approach has been tested in two analytical development projects (1. Machining Daily Demand report

for an industrial enterprise; 2. Population Forecasting dashboard for a healthcare company) (Section IV.C).

III. STATE OF THE ART REGARDING THE CONFIGURATION OF ANALYTICAL PROCESS SUPPORT

A. Relevant characteristics for tools/models

Analytical/Bi applications that can be successfully used in practice are characterized by the fact that they provide 1. the right **information** at 2. the right **time** in 3. a suitable **presentation form** and generated with 4. the right **analysis methods** to 5. the right **users** [5]. The following additional design aspects (identified in a literature review) for tools/models supporting the configuration of analytical applications address the denoted detailed causes for delays or failure of analytical development projects mentioned in Section I.A:

6. **Models utilized by business users** (causes 1 + 2): Requirements documentation tools should be designed for users [23] to get them more involved in the analytical design process.
7. **Graphical modeling notation** (cause 2): Graphical models in requirements documentation [24] provide a more intuitive access to conceptual models [25].
8. **Provision of configuration alternatives** (causes 1 + 2): Providing configuration alternatives in analytical requirements models [26] ensures acceleration of selection decisions and reduces the risk of misleading requirements descriptions.
9. **Data privacy** (cause 3): An examination of the planned analytical use cases from a data privacy perspective is an important task within the requirements analysis to prevent extensive adjustments during the implementation of analytical applications [27].
10. **Process-relation** (cause 4): A strong link to process design in the phase of requirements elicitation and requirements documentation [1] has a positive effect on an analytical information provision that is coordinated between the process activities.

To stress the deep integration of operational processes with analytical applications (accompanied and pushed, e.g., by the dissemination of approaches such as "Business Process Intelligence", "Business Activity Monitoring", "Operational BI" [28] and "Context-Oriented Analytical Applications" [29] in research and practice), further important requirements aspects regarding analytical application design (identified in a literature review) must be added:

11. **Service-oriented design**: To support the adjustment of analytical applications due to changing processes [6], the modularized provision of subcomponents of analytical applications as reusable analytical services in terms of service-oriented architectures (SOA) [4], [30] enables customer-centric service provision [31]

as well as the (re)combination of analytical components from different providers [32].

12. **Self-services**: To enable rapid customization of analytical applications [7] due to changes in process information demand, analytical self-service applications should enable process staff to independently adapt or create analytical reports/dashboards, to integrate new data, to check and/or improve data quality and to adjust analytical data models [33].
13. **Automatic actions**: The proliferation of IoT assets and their integration into operational process controls [34], the acceleration of operational applications (e.g., faster data storage structures) and direct interconnections between systems of data origination and data use are drivers for "real-time enterprises" with the ability to react immediately to occurring events [35]. Business Activity Monitoring (BAM) applications [8] monitor process executions to identify threshold violations or error events and support the automated/rule-based execution of actions.

As a consequence of the aspects listed above, tools/models for the conceptual configuration of analytical applications in order to document user requirements should address these various aspects to improve their usefulness for practice. This is examined in more detail in the following subsections.

B. Scientific models to support the configuration of requirements for analytical applications

The structured literature review (presented in more detail in [15]) with regard to scientific models for analytical requirements documentation and configuration encompassed a very broad search space ("requirements" AND "analytical software" OR "information systems") in order to obtain results without restrictions in the perspectives of observation (e.g., business engineering) and business domains (e.g., production). The analysis of 13 identified requirements and configuration approaches [26], [36]-[47] aimed to identify to what extent these approaches comprehensively ("X"), partially ("(X)"), or do not ("-") address the requirement aspects enumerated in Section III.A. The results of this analysis showed that none of these approaches even comes close to addressing all aspects, with major deficits in the areas „provision of configuration alternatives“, „service-oriented design“, „process-relation“, „periodicity“, „presentation“, „automatic actions“, „self-services“ and „data privacy“ (Table I).

C. Representation of information requirements in process modeling languages

In addition to the modeling approaches just described, which focus on the configuration of analytical applications/requirements, process-modeling languages can describe information requirements from the process design

perspective as well. The standard process modeling languages widespread in practice (e.g., Business Process Model and Notation (BPMN), Extended Event Driven Process Chain (eEPK), UML Activity Diagram) contain the so-called information or data objects, which represent informational inputs in or outputs from process activities in graphical process models. However, the information or data objects in the standard versions of the above-mentioned process notations are only black-box objects unveiling just an identifier without further details (e.g., without content specification, the origin of information or the way of information provision). With these modeling objects, it is not possible to provide a comprehensive specification of a desired information provision [48]. The same abbreviated black-box representation applies to process-relevant databases and software applications.

In addition to the standard versions of process modeling notations, a large number of notational extensions especially for BPMN have emerged (surveyed by [19]). Many of them support the representation of technical and data-oriented content not included in this research. This involves the representation of technical data models (e.g., [49]) and the representation of backend data flows, data changes and technical interactions with data stores (e.g., [49], [50]).

Besides these technical approaches, there are also modeling extensions focused on individual domain-specific requirements aspects, which predominantly do not address the specification of analytical requirements. These BPMN model extensions represent process requirements in specific application domains/industries such as disaster management [51]. The only exception is [52], but this approach considers just a very small part of the user-oriented re-

quirement spectrum of analytical applications with the specification of threshold values for specific key figures to support simulation runs (addressing “data/information” and “automatic actions” (c.f. Table I)).

D. Conceptual configurators for analytical application products

The conceptual configuration of analytical products belongs to the product configuration, which pursues the goal of specifying the quality and structure of product-relevant characteristics [53]. The product configuration distinguishes the customer-inherent configuration (customers/users can select product parameters/characteristics freely) as well as the customer-coherent configuration (customers/users can select predefined parameter sets) [53].

First, a configuration system consists of a configuration component as a content-logical model with configuration elements, configuration variants and their relations. Furthermore, a presentation component allows the interaction between the configuration system user and the configuration component [54]. For the implementation of configuration systems in the special context of software configuration, different kinds of configurators/configuration models are applicable [55]: 1. Software reference models to analyze the potentials of a software product including the description of data structures, operational transactions (functions) and supported processes, 2. checklists for the interactive and systematic reduction of the configuration area regarding a (standard) software product by a question and answer dialogue between user and system, and 3. preconfigured systems as exemplary preselected configuration variants for a homogeneous target group. In the case of a flexible conceptual application configuration by users themselves or in

TABLE I.
COMPARISON OF MODELS SUPPORTING THE CONFIGURATION OF REQUIREMENTS FOR ANALYTICAL APPLICATIONS [15]

	Models utilized by business users	Provision of configuration alternatives	Graphical modeling notation	Service-oriented design	Process-relation	Modeling content for analytical requirements							
						Data / information	Periodicity	Presentation	Users	Analysis methods	Automatic actions	Self-services	Data privacy
[40]	-	-	X	-	-	X	-	-	X	-	-	-	-
[37]	X	-	(X)	-	-	X	-	-	X	-	-	-	-
[41]	(X)	-	X	-	-	X	(X)	(X)	X	(X)	-	-	-
[36]	-	-	-	-	-	X	(X)	-	X	-	-	-	(X)
[38]	-	-	X	-	-	X	-	-	-	-	-	-	-
[39]	-	-	X	-	-	X	-	-	X	-	-	-	-
[42]	X	-	X	-	-	X	-	-	-	-	-	-	-
[26]	X	X	-	-	-	-	-	(X)	X	(X)	-	-	-
[43]	X	-	X	-	-	(X)	-	-	-	-	-	-	-
[44]	(X)	-	(X)	-	-	X	-	-	-	-	-	-	-
[46]	(X)	-	(X)	-	-	X	-	-	-	-	-	-	-
[45]	-	-	X	-	-	X	-	-	-	-	-	-	-
[47]	X	-	X	(X)	(X)	(X)	-	-	X	-	-	-	-

direct interaction with the users, the focus in this current work lies on conceptual configuration models in the form of checklists/requirements catalogs.

The availability of analytical self-service functions within software tools, which could also be regarded as a type of user-sided configuration of analytical applications, is deliberately excluded from this study about the provision of configuration systems in practical products. This is due to the use of analytical self-service functions requires certain in-depth technical or data-related knowledge that cannot be assumed from users of analytical applications (especially casual users like telephone agents in call centers). Furthermore, due to a reduction of complexity, analytical self-service functions represent only a limited part of the actual functional range and available design variants of analytical tools, and thus offer self-service users only limited options for system (re-)design and a comprehensive implementation of their requirements.

The analysis of analytical products with regard to pre-built models/configurators for documenting business requirements (e.g., as a support function for implementation projects) included all product vendors except niche players in the "Gartner Magic Quadrant for Analytics and BI Platforms 2021" [20]: Microsoft, Tableau and Qlik in the "Leaders" quadrant; MicroStrategy, Domo and Google (Looker) in the "Challengers" quadrant; and Sisense, ThoughtSpot, Oracle, SAS, SAP, Yellowfin and TIBCO Software in the "Visionaries" quadrant. Within all examined analytical software products, there are no specific models/configurators to support the collection of user requirements. Taking the example of Microsoft Power BI, available limited support in this area includes the specification of textual change requests regarding existing dashboards/reports via a comment function (plain text without structuring guidelines). Furthermore, in some tools it is possible to integrate external software development applications (e.g., Github) to maintain requirements. To support collaborative development, some vendors establish community areas to collect and discuss ideas for further developments/adaptations of the standard functions of the tools. However, this does not serve to specify/configure concrete requirements for individual use cases.

As an example, TIBCO Software explicitly stated that the provision of models for product-specific application configuration/requirements documentation is deliberately not offered as a part of its own tool. This means that software vendors pass the responsibility and the choice of suitable requirements configurators/documentation models to the external implementation partners.

E. Requirements documentation in the context of analytical development projects

Referring to an own online survey (n=6, Section II provide more information about the research method) about tools and models used in analytical implementation projects

for requirements documentation, Table II shows the mentioned models and documents and their prevalence in practice (column "Sum"). Textual and unstructured/less structured use case descriptions are by far the most widespread tool for requirements documentation. It is worth mentioning here that with use case descriptions, the users with their concrete (usage and operating) requirements have already moved into the center of attention, meanwhile conventional formats such as requirements specification sheets seems to play a minor role. They are followed at some distance by both the content-structured requirements catalogs (checklists) and data models. However, data models are not able to provide even a complete picture of the various requirement facets (e.g., with respect to structural and graphical design of user interfaces, access and distribution paths of information) due to their purely data-oriented view.

TABLE II.
ASSESSMENT OF THE COMPLETENESS OF CONTENT

Models/documents	Very good	Good	Less good	Bad	Sum
Textual user story	1	3	2	-	6
Requirements catalog	1	2	-	-	3
Data model	3	-	-	-	3
Requirements specification sheet	-	-	2	-	2
Backlog	-	1	-	-	1
Process description	-	1	-	-	1
Mockup/PoC	1	-	-	-	1

TABLE III.
ASSESSMENT OF THE COMPREHENSIBILITY OF CONTENT FOR ALL STAKEHOLDERS

Models/documents	Very good	Good	Less good	Bad
Textual user story	-	2	3	1
Requirements catalog	2	1	-	-
Data model	2	1	-	-
Requirements specification sheet	-	-	2	-
Backlog	-	1	-	-
Process description	-	-	1	-
Mockup/PoC	1	-	-	-

As expected and in addition to the one-sided (technical) data models and the mockups (belonging rather to the technical implementation area), requirements catalogs perform best in terms of achievable completeness of requirements content (Table II). This happens because requirements catalogs already mention various design alternatives of an analytical application, which can thus be considered or deliberately excluded from the beginning of application development. Requirements catalogs are also obviously well suited to achieve requirements documentations that are under-

standable both for business users and developers (Table III), since here (in contrast to the lower rated purely textual documents) structural relationships and terminologies are already defined [56] to facilitate the creation of a common understanding. Requirements catalogs also reached a positive score regarding the frequency of inconsistencies (Table IV), because a clear requirements structure in catalogs can be recognized and compared more easily than content in unstructured continuous texts (e.g., use case descriptions).

TABLE IV.
ASSESSMENT OF THE FREQUENCY OF INCONSISTENCIES WITH
OTHER MODELS/DOCUMENTS USED FOR REQUIREMENTS
DOCUMENTATION

Models/documents	Very often	Often	Rather often	Less often	Rarely
Textual user story	2	2	1	1	-
Requirements catalog	-	-	-	-	3
Data model	-	-	-	-	3
Requirements specification sheet	1	-	-	1	-
Backlog	-	-	1	-	-
Process description	-	1	-	-	-
Mockup/PoC	-	-	-	-	1

A similar result emerges how data privacy risks are taken into account in requirements documentation (Table V): Text-based documents got a worse score here, while requirement catalogs at least received a better rating. It is remarkable that no model/document was able to achieve a very good rating. This again substantiates the still inadequate consideration of data privacy risks in requirements documentation.

One expert in this survey provided detailed information

TABLE V.
ASSESSMENT OF THE CONSIDERATION OF DATA PRIVACY RISKS

Models/documents	Very good	Good	Less good	Bad
Textual user story	-	2	3	1
Requirements catalog	-	2	1	-
Data model	-	1	-	2
Requirements specification sheet	-	-	1	1
Backlog	-	-	-	1
Process description	-	1	-	-
Mockup/PoC	-	1	-	-

about the specific requirements catalog models used in projects. These are the cross-domain requirements templates according to [56] to create requirements in form of sentences with specific content placeholders in a particular order, which ensures a uniform way of formulating requirements. Unfortunately, this predefined formulation structure alone does not provide any information about the structural and

content design of domain-specific applications and the relevant analytical requirements aspects and variants.

Based on these results, requirements catalogs seem to be best suited for documenting requirements with regard to the design of analytical applications from a practical point of view in terms of a complete, consistent and unambiguous provision of content. Furthermore, a new requirements catalog for analytical applications should encompass the specific functional and non-functional properties of this software domain, as well as actively consider the other model aspects described in Section III.A.

IV. ANALYTICAL SERVICE MODELS FOR THE CONCEPTUAL CONFIGURATION OF ANALYTICAL APPLICATIONS

A. Presentation of the modeling approach

The configuration approach already presented in more detail in [15] enables the configuration of analytical services to document requirements in process contexts. The use of services in the sense of encapsulated functions connected via standardized interfaces [3] allows the flexible conceptual (re-)configuration of analytical applications. The configuration approach is appropriate for different analytical use cases, business domains, data formats as well as analysis methods. It can be classified as a customer-inherent product configuration [53] in order to permit a modular orchestration of analytical components/services.

The configuration approach is based on the distinction of three different configuration areas (cf. [15]):

- Use Case-Specific Configuration Content enables the specification of individual reports/dashboards.
- Configuration Content for Analysis Preparation describes the required functional scope in the area of analytical self-services.
- Use Case-Overlapping Configuration Content addresses design aspects that affect the entire analytical application.

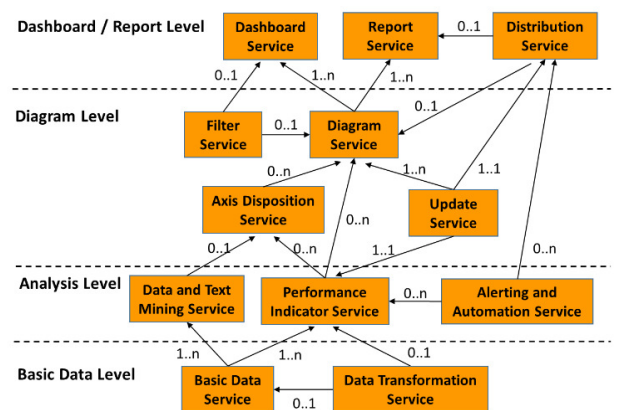


Fig.1 Analytical service types and their interconnections in the configuration area Use Case-Specific Configuration Content [15]

Each configuration area contains a set of analytical service types, which can be configured in more detail (c.f. Fig. 4-7) to describe specific aspects of the analytical application more precisely. Fig. 1 shows the service network for the Use Case-Specific Configuration Content with its associated analytical service types on four levels and their logical interconnections. In this way it describes reports/dashboards, their diagrams as well as the key figures displayed therein and the underlying basic data.

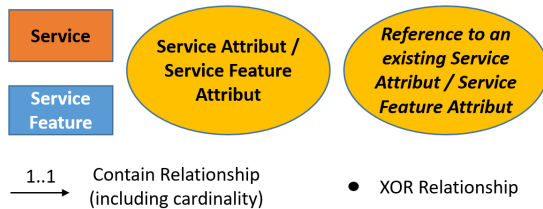


Fig. 2 Modeling objects for the internal configuration of analytical services [15]

Fig. 2 shows the different modeling objects available for the graphical configuration of the specific service features within the individual analytical services (c.f. Fig. 4-7). In order to be able to represent the interconnection of the analytical services needed to configure a specific analytical use case (in terms of a specific dashboard or report) in connection with the related process activity, the analytical services are represented as data objects (e.g., Fig. 3) in BPMN pro-

cess models. To make the respective analytical service types distinguishable, the BPMN data objects bear specific identifiers (c.f. [15]).

B. Results of the expert reviews

Within the intensive review sessions (c.f. Section II), the analytical experts got an overview about the whole configuration approach for analytical services (Section IV.A) concerning the three specific interconnected analytical service type networks (Fig. 1 and [15]) within the three different configuration areas (Section IV.A), and about the numerous service type-specific graphical service feature configurations (e.g., Fig. 4-7). As a result, the experts confirmed that the configuration approach has a predominantly high and increased potential benefit for practice (Table VI). This concerns in particular the areas of unambiguity and completeness of content, modularization of requirements and the possibility of reuse, the identification of new design options, obtaining an overview about analytical process support and using the instantiated models as a starting point for deriving a technical concept. It is also noteworthy that this approach was rated with a predominantly higher added value compared to models previously used in these companies.

C. Presentation of evaluation use cases

Based on the positive evaluation by the experts (Section IV.B), the configuration approach was tested in a healthcare project to develop a dashboard presenting annual **Popula-**

TABLE VI.

ASSESSMENT OF THE POTENTIAL BENEFITS OF THE ANALYTICAL SERVICES CONFIGURATION APPROACH FOR REQUIREMENTS DOCUMENTATION

	High benefit	Increased benefit	Less high benefit	Low benefit	No benefit
Unambiguity of content	9	3	1	-	-
Completeness of content	9	4	-	-	-
Provision of selectable design variants	4	2	2	1	-
Saving of effort in practical projects	3	7	1	2	-
Modular and simple (re-)combination of requirement contents / services	5	4	-	-	-
Recording data privacy-specific conditions of basic data	2	3	2	2	-
Specification of requirements for analytical self-service functions	4	2	1	-	-
Starting point for deriving a technical concept	9	4	-	-	-
Identification of design options not yet considered through proposed configuration content	5	2	-	-	-
Starting point for subsequent adaptations of the analytical software at the same customer	6	5	2	-	-
Starting point for similar future projects with other customers	6	7	-	-	-
Obtaining an overview of the analytical process support	7	6	-	-	-
Representing the sequence of analytical content in processes	5	5	3	-	-
Representing relationships between the individual analytical services via associated BPMN data objects	7	5	1	-	-
	High added value	Increased added value	Less high added value	Low added value	No added value
Added value of the configuration approach to requirements documentation compared to the previous approach in the company	3	7	2	1	-
Added value of the coupled representation of analytical support and user processes compared to the previous requirements documentation in the company	2	8	2	-	-

tion Forecast information in different counties in the federal state of Saxony. In this project, some planning for the dashboard design had already been done before. Based on this, all analytical services needed to describe the entire analytical use case (from the basic data to the key figure calculation and visualization up to the integration in the dashboard) were configured together with three development team members (project manager, data analyst, analytical developer). Due to the numerous design features and design variants considered in the different analytical services, some new facets and features of the dashboard that had not yet been considered in the previous development were identified. The elaborated analytical service models subsequently served as the conceptual basis for further dashboard implementation. The positive feedback from the three experts involved (high benefit: 16 times; increased benefit: 16 times; less high benefit: 3 times; low benefit: 1 time; willingness to use in future projects: 3 out of 3) (Table VI) was clear evidence of the usefulness, acceptance and, in particular, practicability of the configuration approach in implementation projects.

In a second case study, the configuration approach was used to configure a **Machining Daily Demand** report in an industrial enterprise. This case study was not developed in the middle of an ongoing project with analytical experts, but rather at the beginning of the requirements elicitation process together with a requirements provider from the business department. This report should inform a storekeeper in near real-time which parts from stock are needed in which quantities, according to the current planning for the supply of the daily production. To obtain an overview, Fig. 3 shows the process model for this use case with the interconnections of all involved analytical services.

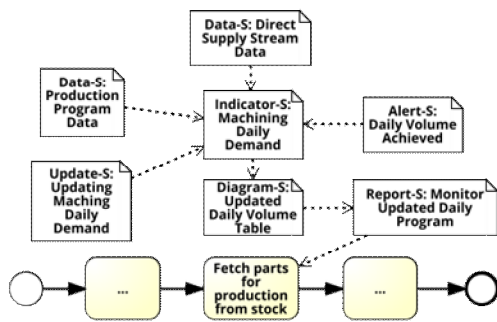


Fig. 3 Process model of the use case Machining Daily Demand

The "Performance Indicator Service" (Indicator-S) (Fig. 4) plays a central role in the use case. The calculation of the indicator "Machining Daily Demand" should not only take into account the "Basic Data Service" (Data-S) "Production Program Data" provided at the beginning of a day's production, but also consider events in the upstream parts flow and the resulting dynamic changes for the supply of the other

parts ("Direct Supply Stream Data" (Fig. 5); c.f. service interconnections in Fig. 3).

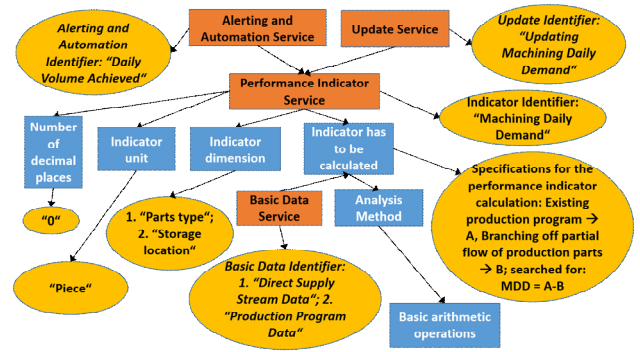


Fig. 4 "Performance Indicator Service" "Machining Daily Demand"

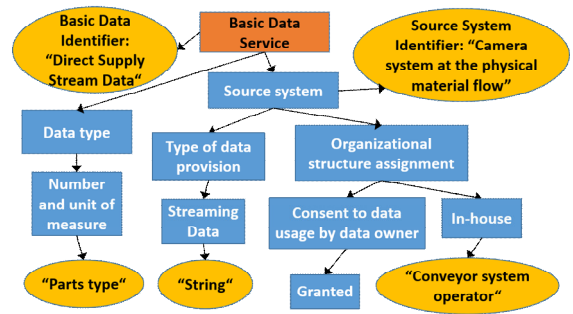


Fig. 5 "Basic Data Services" "Direct Supply Stream Data"

The "Alerting and Automation Service" (Alert-S) defines to monitor the development of the indicator and to trigger an automatic on-screen warning message and a control instruction to a robot if the threshold is exceeded (Fig. 6). Finally, the "Report Service" specifies the access conditions for the prospected user role (storekeeper) (Fig. 7).

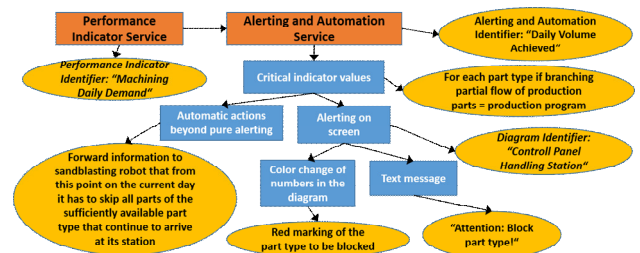


Fig. 6 "Alerting and Automation Service" "Daily Volume Achieved"

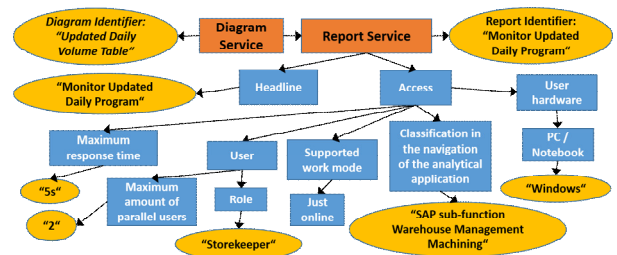


Fig. 7 "Report Service" "Monitor Updated Daily Program"

V. CONCLUSION

Based on identified challenges in practice for the documentation of process-related requirements in the context of analytical software development in combination with challenges emerging with process-oriented analytics, this work has shown that neither scientific approaches (requirements models for analytical applications; process modeling languages and their language extensions), analytical products nor the textual documents and models predominantly used in implementation projects for requirements documentation are able to sufficiently address these challenges and the essential analytical design aspects. Furthermore, evidence was provided by experts that a service-based conceptual approach for the customer-inherent [53] configuration of analytical services [15] addressing all 13 requirements aspects relevant for analytical application development (Section II-I.A) leads to a predominantly increased to high benefit for analytical development projects.

The applied research design did not comprehensively investigate all facets of this topic, and this leaves opportunities for further research. A more extensive investigation of configuration models used in analytical implementation projects could yield additional design recommendations for the further development of the configuration approach. Furthermore, in the current approach, data objects with a label of the service type and a unique identifier represents the analytical services in the process models. Future research activities could investigate which essential service features from the detailed service models could expand the content to be represented in the data objects (using extension mechanisms for type definitions available in BPMN) to visualize more detailed information about analytical process support directly in the process models.

REFERENCES

- [1] A. D. N. Sarma, "A Generic Functional Architecture for Operational BI System," *International Journal of Business Intelligence Research*, vol. 9, no. 1, pp. 64–77, 2018, doi: 10.4018/IJBIR.2018010105.
- [2] T. Hänel and C. Felden, "Operational Business Intelligence im Zukunftsszenario der Industrie 4.0," in *Analytische Informationssysteme: Business Intelligence-Technologien und -Anwendungen*, P. Gluchowski and P. Chamoni, Eds., 5th ed.: Springer Verlag, 2016, pp. 259–282.
- [3] P. Alpar, R. Alt, F. Bensberg, and P. Weimann, *Anwendungsorientierte Wirtschaftsinformatik: Strategische Planung, Entwicklung und Nutzung von Informationssystemen*, 9th ed. Wiesbaden: Springer Vieweg, 2019.
- [4] Z. Panian, "How to Make Business Intelligence Actionable through Service-oriented Architectures," in *2nd WSEAS International Conference on Computer Engineering and Applications*, 2008, pp. 210–221.
- [5] K. D. Schulze and C. Dittmar, "Business Intelligence Reifegradmodelle," in *Analytische Informationssysteme: Business Intelligence-Technologien und -Anwendungen*, P. Chamoni and P. Gluchowski, Eds., 3rd ed., Berlin: Springer Verlag, 2006, pp. 72–87.
- [6] E. Colangelo and T. Bauernhansl, "Usage of Analytical Services in Industry Today and Tomorrow," *Procedia CIRP*, vol. 57, pp. 276–280, 2016, doi: 10.1016/j.procir.2016.11.048.
- [7] A. Hoffjan and M. Rohe, "5. Konzeptionelle Analyse von Self-Service Business Intelligence und deren Gestaltungsmöglichkeiten," in *Erfolgreiches Controlling*, M. Kibler and A. Wieseahn, Eds.: Nomos Verlagsgesellschaft mbH & Co. KG, 2018, pp. 99–112.
- [8] A. Vera-Baquero, R. Colomo-Palacios, and O. Molloy, "Real-time business activity monitoring and analysis of process performance on big-data domains," *Telematics and Informatics*, vol. 33, no. 3, pp. 793–807, 2016, doi: 10.1016/j.tele.2015.12.005.
- [9] M. Begerow, *Ziele von Business Intelligence*. [Online]. Available: <https://datenbanken-verstehen.de/business-intelligence/business-intelligence-grundlagen/business-intelligence-ziele/> (accessed: Nov. 9 2020).
- [10] *Warum scheitern viele BI-Projekte?* [Online]. Available: <https://www.aep-ag.com/2-uncategorised/211-warum-scheitern-viele-bi-projekte> (accessed: Nov. 9 2020).
- [11] *Die sechs häufigsten Fehler in Business Intelligence Projekten*. [Online]. Available: <https://www.prisma-informatik.de/erp-blog/2016/06/die-sechs-haeufigsten-fehler-in-business-intelligence-projekten/> (accessed: Nov. 9 2020).
- [12] D. Meister, "Woran scheitern Data Science Projekte?," *Datahouse AG*, 2019.
- [13] O. Fleming, T. Fountaine, N. Henke, and T. Saleh, *Ten red flags signaling your analytics program will fail*. [Online]. Available: <https://www.mckinsey.com/business-functions/mckinsey-analytics/our-insights/ten-red-flags-signaling-your-analytics-program-will-fail> (accessed: Nov. 23 2020).
- [14] P. Uria-Recio, *Top 25 Mistakes Corporates Make in their Advanced Analytics Programs*. [Online]. Available: <https://towardsdatascience.com/top-25-mistakes-corporates-make-in-their-advanced-analytics-programs-c51e76218e20> (accessed: Jun. 15 2021).
- [15] C. Hrach and R. Alt, "Configuration Approach for Analytical Service Models – Development and Evaluation," in *2020 IEEE 22nd Conference on Business Informatics (CBI)*, Antwerp, Belgium, 2020, pp. 260–269.
- [16] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, "A Design Science Research Methodology for Information Systems Research," *Journal of Management Information Systems*, vol. 24, no. 3, pp. 45–77, 2007, doi: 10.2753/MIS0742-1222240302.
- [17] *Business Process Model and Notation: An introductory guide*. [Online]. Available: <https://www.signavio.com/bpmn-introductory-guide/> (accessed: Dec. 30 2020).
- [18] T. Allweyer, *BPMN setzt sich durch in der Praxis*. [Online]. Available: <https://www.computerwoche.de/a/bpmn-setzt-sich-durch-in-der-praxis,1886445> (accessed: Dec. 30 2020).
- [19] K. Zarour, D. Benmerzoug, N. Guermouche, and K. Drira, "A systematic literature review on BPMN extensions," *BPMJ*, vol. 26, no. 6, pp. 1473–1503, 2019, doi: 10.1108/BPMJ-01-2019-0040.
- [20] J. Richardson, K. Schlegel, R. Sallam, A. Kronz, and J. Sun, "Magic Quadrant for Analytics and Business Intelligence Platforms 2021," 2021.
- [21] S. Gregor and A. R. Hevner, "Positioning and Presenting Design Science Research for Maximum Impact," *MISQ*, vol. 37, no. 2, pp. 337–355, 2013, doi: 10.25300/MISQ/2013/37.2.01.
- [22] G. B. Davis, "Advising and Supervising," in *Butterworth-Heinemann information systems series, Research in information systems: A handbook for research supervisors and their students*, D. E. Avison and J. Pries-Heje, Eds., Amsterdam: Elsevier Butterworth-Heinemann, 2005, pp. 1–33.
- [23] O. Liskin, "How Artifacts Support and Impede Requirements Communication," in *Lecture notes in computer science, Requirements Engineering: Foundation for Software Quality*, S. A. Fricker and K. Schneider, Eds., Cham: Springer International Publishing, 2015, pp. 132–147.
- [24] J. Misra, S. Sengupta, and S. Podder, "Topic cohesion preserving requirements clustering," in *Proceedings of the 5th International Workshop on Realizing Artificial Intelligence Synergies in Software Engineering - RAISE '16*, Austin, Texas, 2016, pp. 22–28.
- [25] U. Frank, "Domain-Specific Modeling Languages: Requirements Analysis and Design Guidelines," in *Domain engineering: Product lines, languages, and conceptual models*, I. Reinhartz-Berger, A. Sturm, T. Clark, S. Cohen, and J. Bettin, Eds., Heidelberg: Springer Verlag, 2013, pp. 133–157.
- [26] J. H. Mayer, R. Winter, and T. Mohr, "Situational Management Support Systems," *Bus Inf Syst Eng*, vol. 4, no. 6, pp. 331–345, 2012, doi: 10.1007/s12599-012-0233-5.

- [27] S. Sharma, K. Chen, and A. Shet, "Towards Practical Privacy-Preserving Analytics for IoT and Cloud-Based Healthcare Systems," *IEEE Internet Computing*, March-April, 2018, doi: 10.1109/MIC.2018.112102519.
- [28] E. Graupner, M. Berner, A. Mädche, and H. Jegadeesan, "Business Intelligence & Analytics for Processes \textendash A Visibility Requirements Evaluation," in *Multikonferenz Wirtschaftsinformatik (MKWI)*, Paderborn, Germany, 26. - 28. Februar 2014. Ed.: D. Kundisch, 2014, pp. 154–166.
- [29] H.-G. Kemper, H. Baars, and W. Mehanna, *Business Intelligence – Grundlagen und praktische Anwendungen*, 3rd ed. Wiesbaden: Vieweg+Teubner, 2010.
- [30] J. Schiefer and A. Seufert, "Towards a Service-Oriented Architecture for Operational BI," in *Multikonferenz Wirtschaftsinformatik 2010*, 2010, pp. 1137–1149.
- [31] S. Sachse, "Customer-centric Service Management - Conceptualization and Evaluation of Consumer-induced Service Composition," Dissertation, Institut für Wirtschaftsinformatik, Universität Leipzig, Leipzig, 2018.
- [32] L. Wu, G. Barash, and C. Bartolini, "A Service-oriented Architecture for Business Intelligence," in *IEEE International Conference on Service-Oriented Computing and Applications (SOCA '07)*, 2007, pp. 279–285.
- [33] P. Alpar and M. Schulz, "Self-Service Business Intelligence," *Bus Inf Syst Eng*, vol. 58, no. 2, pp. 151–155, 2016, doi: 10.1007/s12599-016-0424-6.
- [34] S. Schöning, S. Jablonski, and A. Ermer, "IoT-basiertes Prozessmanagement," *Informatik Spektrum*, vol. 42, no. 2, pp. 130–137, 2019, doi: 10.1007/s00287-019-01140-x.
- [35] D. Beverungen et al., "Seven Paradoxes of Business Process Management in a Hyper-Connected World," *Bus Inf Syst Eng*, vol. 18, no. 2, p. 279, 2020, doi: 10.1007/s12599-020-00646-z.
- [36] M. Goeken, "Anforderungsmanagement bei der Entwicklung von Data Warehouse-Systemen," in *Auf dem Weg zur Integration Factory: Proceedings der DW2004 - Data Warehousing und EAI*, J. Schelp and R. Winter, Eds., Heidelberg: Physica Verlag, 2004, pp. 167–186.
- [37] A. Bonifati, F. Cattaneo, S. Ceri, A. Fuggetta, S. Paraboschi, and P. Di Milano, "Designing Data Marts for Data Warehouses," *ACM Transactions on Software Engineering and Methodology*, vol. 10, pp. 452–483, 2001, doi: 10.1145/384189.384190.
- [38] D. Calvanese, L. Dragone, D. Nardi, R. Rosati, and S. M. Trisolini, "Enterprise modeling and Data Warehousing in Telecom Italia," *Information Systems*, vol. 31, no. 1, pp. 1–32, 2006, doi: 10.1016/j.is.2004.07.002.
- [39] P. Giorgini, S. Rizzi, and M. Garzetti, "GRAnD: A goal-oriented approach to requirement analysis in data warehouses," *Decision Support Systems*, vol. 45, no. 1, pp. 4–21, 2008, doi: 10.1016/j.dss.2006.12.001.
- [40] G. Shanks and P. Darke, "Understanding corporate data models," *Information & Management*, vol. 35, no. 1, pp. 19–30, 1999, doi: 10.1016/S0378-7206(98)00078-0.
- [41] B. Strauch, "Entwicklung einer Methode für die Informationsbedarfsanalyse im Data Warehousing," Universität St. Gallen, St. Gallen, 2002.
- [42] A. Maté and J. Trujillo, "A trace metamodel proposal based on the model driven architecture framework for the traceability of user requirements in data warehouses," *Information Systems*, vol. 37, no. 8, pp. 753–766, 2012, doi: 10.1016/j.is.2012.05.003.
- [43] J. Horkoff et al., "Strategic business modeling: representation and reasoning," *Softw Syst Model*, vol. 13, no. 3, pp. 1015–1041, 2014, doi: 10.1007/s10270-012-0290-8.
- [44] P. Jovanovic, O. Romero, A. Simitis, A. Abelló, and D. Mayorova, "A requirement-driven approach to the design and evolution of data warehouses," *Information Systems*, vol. 44, pp. 94–119, 2014, doi: 10.1016/j.is.2014.01.004.
- [45] C. Rosenkranz, R. Holten, M. Räkers, and W. Behrmann, "Supporting the design of data integration requirements during the development of data warehouses: a communication theory-based approach," *European Journal of Information Systems*, vol. 26, no. 1, pp. 84–115, 2017, doi: 10.1057/ejis.2015.22.
- [46] A. Ferrández, A. Maté, J. Peral, J. Trujillo, E. de Gregorio, and M.-A. Aufaure, "A framework for enriching Data Warehouse analysis with Question Answering systems," *J Intell Inf Syst*, vol. 46, no. 1, pp. 61–82, 2016, doi: 10.1007/s10844-014-0351-2.
- [47] M. A. Teruel, A. Maté, E. Navarro, P. González, and J. C. Trujillo, "The New Era of Business Intelligence Applications: Building from a Collaborative Point of View," *Bus Inf Syst Eng*, vol. 61, no. 5, pp. 615–634, 2019, doi: 10.1007/s12599-019-00578-3.
- [48] M. O'Shea, G. Pawellek, and A. Schramm, "Durch maßgeschneiderte Informationsversorgung zu mehr Usability," *Wirtschaftsinformatik & Management (WuM)*, vol. 5, no. 6, pp. 104–113, 2013, doi: 10.1365/s35764-013-0370-8.
- [49] P. Bocciarelli, A. D'Ambrogio, E. Paglia, and A. Giglio, "An HLA-based BPMN extension for the specification of business process collaborations," in *2017 IEEE/ACM 21st International Symposium on Distributed Simulation and Real Time Applications (DS-RT)*, Rome, 2017, pp. 1–8.
- [50] M. Magnani and D. Montesi, "BPDMN: A Conservative Extension of BPMN with Enhanced Data Representation Capabilities," 2009. [Online]. Available: <http://arxiv.org/pdf/0907.1978v1>
- [51] H. Betke and M. Seifert, "BPMN for Disaster Response Processes," in *INFORMATIK 2017*, Chemnitz, 2017, pp. 1311–1324.
- [52] A. D'Ambrogio, E. Paglia, P. Bocciarelli, and A. Giglio, "Towards performance-oriented perfective evolution of BPMN models," in *2016 Symposium on Theory of Modeling and Simulation (TMS-DEVS)*, 2016, pp. 1–8.
- [53] T. Blecker, H. Dullnig, and F. Malle, "Kundenkohärente und kundenhärente Produktkonfiguration in der Mass Customization," *Industrie Management*, vol. 19, no. 1, pp. 21–24, 2003.
- [54] F. T. Piller, *Mass Customization: Ein wettbewerbsstrategisches Konzept im Informationszeitalter*. Zugl.: Würzburg, Univ., Diss., 1999 u.d.T.: Kundenindividuelle Massenproduktion (mass customization) als wettbewerbsstrategisches Modell industrieller Wertschöpfung in der Informationsgesellschaft, 4th ed. Wiesbaden: Dt. Univ.-Verl., 2006.
- [55] J. Ritter, "Prozessorientierte Konfiguration komponentenbasierter Anwendungssysteme," Dissertation, Universität Oldenburg, Oldenburg, 2000.
- [56] C. Rupp, *Requirements-Engineering und -Management: Das Handbuch für Anforderungen in jeder Situation*, 7th ed. München: Hanser, 2021.

A novel iterative approach to determining compromise rankings

Bartłomiej Kizielewicz*, Andrii Shekhovtsov, Wojciech Sałabun

Research Team on Intelligent Decision Support Systems,
Department of Artificial Intelligence Methods and Applied Mathematics,
Faculty of Computer Science and Information Technology
West Pomeranian University of Technology in Szczecin
ul. Żołnierska 49, 71-210 Szczecin, Poland

Email: {bartlomiej-kizielewicz, andrii-shekhovtsov, wojciech.salabun@zut}@zut.edu.pl

Abstract—In many cases involving multi-criteria decision-making, we need compromise solutions. This is a crucial aspect due to the specific characteristics of decision problems. However, the proposed compromise approaches are often complex to verify to what extent they are reliable. Therefore, this paper proposes a new iterative approach based on decision option evaluations from selected multi-criteria decision-making methods, i.e., TOPSIS, VIKOR, and SPOTIS. The obtained results have high similarity among each other, which was measured by Spearman’s weighted correlation coefficient and WS ranking similarity coefficient. Furthermore, the proposed approach showed high efficiency and adaptability of the generated results.

I. INTRODUCTION

MANY works propose new approaches related to the topic of multi-criteria decision making, for which conflicting results are obtained compared with classical approaches. In these methods there are a number of parameters that have a great influence on the final evaluations of decision options [1]. Sałabun et al. investigated the effect of normalization and weight selection methods on the final evaluations in selected methods [2]. Ghaleb et al. evaluated the TOPSIS, AHP, and VIKOR methods based on five selected factors [3]. Baydas et al. used the stock price from the finance domain problem as a tool to compare the MCDM methods [4].

Therefore, researchers focus on improving their methods based on compromise solutions. Liao et al. presented an improved SMAA-CO method designed to determine compromise solutions [5]. The approach they presented is characterized by its ability to reflect uncertain preferences of decision-makers with conflicting criteria. Stevic et al. proposed a new compromise MARCOS approach based on ideal solutions and the utility function [6]. The method performed well in a balanced provider selection problem in the healthcare domain, where it remained stable with a large dataset.

A popular approach is voting algorithms to establish a compromise ranking based on reference rankings. One such example is the use of Borda’s approach and Copeland’s method for evaluating the performance of electric vehicle batteries [7]. Approaches based on the Borda and Copeland algorithms have also been used to create a recommendation system based on e-commerce [8] and to select online services [9].

However, these methods mainly focus on rankings in which it is difficult to observe slight differences between the obtained preferences of decision options [10]. Moreover, using methods that suffer from the problem of rank reversal paradox, it is difficult to determine the most compromise ranking.

In this paper, we propose a new method for determining compromise rankings based on reference evaluations. The evaluations of the decision alternatives obtained by the selected methods are used to determine the compromise rankings. The obtained ratings are formed into a decision matrix, where the types of attributes for the newly formed matrix depend on the ranking method. Three multi-criteria decision-making methods such as TOPSIS, VIKOR, and SPOTIS were used in this study.

The main contribution of our work is the concept of a new approach to verifying compromise solutions. Due to the number of existing MCDM approaches, it is necessary to develop compromise methods to produce consistent evaluations and rankings. Therefore, our proposed concept of a compromise approach aims to show the possibility of iteratively determining a compromise ranking based on reference rankings obtained for the original decision matrix.

The remainder of the paper is organized as follows. Section 2 presents descriptions of the TOPSIS, VIKOR, and SPOTIS methods and ranking similarity coefficients. Section 3 presents research on the proposed compromise approach. Finally, Section 4 presents a summary and outlines future research directions.

II. PRELIMINARIES

A. The TOPSIS Method

Technique of Order Preference Similarity (TOPSIS) is based on the ideal solution approach for solving multi-criteria decision problems [11]. The approach evaluates decision alternatives for the distance from a positive ideal solution and a negative ideal solution. TOPSIS is a continuously evolving method capable of solving problems involving uncertain environments [12]. Its basic version can be presented in the following steps:

Step 1. This step includes the determination of a normalized decision matrix. In the vector method used in the presented

work, the square root of all values is calculated. Equations used for profit (1) and cost criteria (2) are presented below.

$$r_{ij} = \frac{x_{ij} - \min(x_j)}{\max(x_j) - \min(x_j)} \quad (1)$$

$$r_{ij} = \frac{\max(x_j) - x_{ij}}{\max(x_j) - \min(x_j)} \quad (2)$$

Step 2. Computation of the weighted values of the normalized decision matrix v_{ij} in accordance with Equation (3).

$$v_{ij} = w_i r_{ij} \quad (3)$$

Step 3. Calculation of the positive ideal solution (PIS) and negative anti-ideal solution (NIS) vectors. The PIS represented by (4) contains the maximum values for every criterion, and the NIS expressed by (5) includes minimum values. It is unnecessary to split the criteria into benefit and cost since the normalization procedure converted the cost criteria to profit criteria.

$$v_j^+ = \{v_1^+, v_2^+, \dots, v_n^+\} = \left\{ \max_j(v_{ij}) \right\} \quad (4)$$

$$v_j^- = \{v_1^-, v_2^-, \dots, v_n^-\} = \left\{ \min_j(v_{ij}) \right\} \quad (5)$$

Step 4. Calculation of distance from PIS by (6) and NIS, by (7) for every alternative under consideration [2].

$$D_i^+ = \sqrt{\sum_{j=1}^n (v_{ij} - v_j^+)^2} \quad (6)$$

$$D_i^- = \sqrt{\sum_{j=1}^n (v_{ij} - v_j^-)^2} \quad (7)$$

Step 5. Calculation of the result for every considered variant by (8). The score ranges from 0 to 1. An alternative that has a preference value closer to 1 is better.

$$C_i = \frac{D_i^-}{D_i^- + D_i^+} \quad (8)$$

B. The VIKOR Method

ViseKriterijumska Optimizacija I Kompromisno Resenje (VIKOR) is a method based on the compromise approach, which evaluates alternatives with conflicting types of criteria [13]. The compromise solution in this method is considered to be the solution that is closest to the ideal. On the other hand, compromise is achieved through mutual concessions. This method can be presented in the following steps:

Step 1. Determination of the best f_j^* and the worst f_j^- value for the function of individual criteria. For benefit criteria, formula (9) is performed

$$f_j^* = \max_i f_{ij}, \quad f_j^- = \min_i f_{ij} \quad (9)$$

while for the cost criteria, formula presented below is applied (10).

$$f_j^* = \min_i f_{ij}, \quad f_j^- = \max_i f_{ij} \quad (10)$$

Step 2. Calculation of S_i and R_i using formulas (11) and (12).

$$S_i = \sum_{j=1}^n w_j \frac{(f_j^* - f_{ij})}{(f_j^* - f_j^-)} \quad (11)$$

$$R_i = \max_j w_j \frac{(f_j^* - f_{ij})}{(f_j^* - f_j^-)} \quad (12)$$

Step 3. Calculation of Q_i using Equation (13).

$$Q_i = v \frac{(S_i - S^*)}{(S^- - S^*)} + (1 - v) \frac{(R_i - R^*)}{(R^- - R^*)} \quad (13)$$

where

$$S^* = \min_i S_i, \quad S^- = \max_i S_i$$

$$R^* = \min_i R_i, \quad R^- = \max_i R_i$$

v means the weight adopted for the strategy of "most criteria". In the calculations in this study v equal to 0.5 was set.

Step 4. Graded variants in S , R and Q are ordering ascending. The outcome is provided in 3 ranked lists.

Step 5. The consensus is suggested regarding the good advantage and acceptable stability concerning vectors received in the preceding stage. The most favourable variant has the lowest value and is the leader of the ranking Q .

C. The SPOTIS Method

Stable Preference Ordering Towards Ideal Solution (SPOTIS) is a newly developed approach robust to the reverse ranking paradox. The approach is based on evaluation concerning the distance from the Ideal Solution Point of the given decision alternatives [14]. Additionally, it allows the introduction of an expert point against which the alternatives are evaluated. It can be presented as follows:

Step 1. Determine the normalized distances computed for Ideal Solution Point as Equation (14) demonstrates.

$$d_{ij}(A_i, S_j^*) = \frac{|S_{ij} - S_j^*|}{|S_j^{max} - S_j^{min}|} \quad (14)$$

Step 2. Calculate weighted normalized distances represented by $d(A_i, S^*) \in [0, 1]$ as Equation (15) shows.

$$d(A_i, S^*) = \sum_{j=1}^N w_j d_{ij}(A_i, S_j^*) \quad (15)$$

The resulting ranking calculated according to $d(A_i, S^*)$ values. Alternatives with lower values of $d(A_i, S^*)$ receive better positions in the ranking. The technique presented in this paper can be demonstrated by an alternative algorithm, included in [14]. However, the authors of this work provided and applied this option because it seems straightforward. Moreover, both versions supply identical outcomes.

D. Similarity coefficients

Ranking similarity coefficients such as the weighted Spearman coefficient r_w and the ranking similarity coefficient WS were used to examine the similarity of the rankings [15], [16]. In the case of the weighted Spearman coefficient, it was designed to reflect the most relevant alternatives that were rated the best. In contrast, the ranking similarity coefficient WS is an asymmetric ranking similarity coefficient where the alternatives at the top are given the most consideration. These coefficients can be represented by the formulas (16) and (17).

$$r_w = 1 - \frac{6 \cdot \sum_{i=1}^n (x_i - y_i)^2 ((N - x_i + 1) + (N - y_i + 1))}{n \cdot (n^3 + n^2 - n - 1)} \tag{16}$$

$$WS = 1 - \sum_{i=1}^n \left(2^{-x_i} \frac{|x_i - y_i|}{\max\{|x_i - 1|, |x_i - N|\}} \right) \tag{17}$$

III. STUDY CASE

Taking advantage of compromise ranking methods is an important topic that often arises in multi-criteria decision-making. We can determine the most flexible ranking under multiple conflicting criteria with them. In the following, we propose a new iterative approach based on the preferences of reference MCDM methods. The whole procedure can be described as follows:

Step 1. Evaluation of the formed decision matrix n by MCDM methods

Step 2. Create a decision matrix based on the ratings of the MCDM methods. Criteria types are determined based on the method's ranking type.

Step 3. Return to step 1 until the ranking $i - 1$ and the ranking i of the selected methods are the same, where ranking $i - 1$ denotes the ranking obtained from the evaluations entering the current decision matrix, and ranking i denotes the currently created ranking.

A decision matrix whose values were randomly generated from a uniform distribution $[0, 1]$ was used for the study. It is created from 4 criteria and 10 alternatives. For each of the considered criteria, the same weight was assigned, and it was assumed that the first two criteria are profit type while the last two criteria are cost type. Finally, the created decision matrix is presented using the table I.

TABLE I: Generated decision matrix consisting of alternatives $A_1 - A_{10}$ and criteria $C_1 - C_4$.

A_i	C_1	C_2	C_3	C_4
A_1	0.989490	0.592018	0.818605	0.031060
A_2	0.083740	0.507472	0.378180	0.976184
A_3	0.445502	0.029106	0.196421	0.897797
A_4	0.408798	0.982134	0.428897	0.126954
A_5	0.471495	0.042197	0.154756	0.379112
A_6	0.205953	0.113477	0.537154	0.396152
A_7	0.481553	0.793211	0.541687	0.265333
A_8	0.515628	0.270621	0.392020	0.281762
A_9	0.789527	0.050453	0.277638	0.179735
A_{10}	0.507601	0.545130	0.644899	0.954265

The obtained preferences of the different alternatives for each iteration are shown using the illustration 1, where 1a represents the preference for the TOPSIS approach, 1b represents the preference for the VIKOR approach, and 1c represents the preference for the SPOTIS approach.

For the TOPSIS approach, stabilization of ratings was faster than for the rest of the methods. For the TOPSIS approach, the range relative to the evaluated decision options increased. The first ratings obtained were in the range $[0.33748, 0.65286]$, while the last ratings were $[0, 1]$. Additionally, the spread of the obtained ratings also changed, where in the first iteration, the standard deviation was 0.11151, while in the last iteration, it was 0.354886.

The stabilization of SPOTIS approach ratings was slightly faster. As with the TOPSIS approach, the range of obtained ratings for subsequent iterations in the VIKOR approach increased. For the first iteration, the obtained preference values were in the range $[0.03838, 1]$, while for the last iteration, the values were in the range $[0, 1]$. The change in scatter of these values, on the other hand, is not as prominent as for the TOPSIS approach. Indeed, for the VIKOR approach, a standard deviation value of 0.32609 was obtained for the first iteration and 0.35500 for the last iteration.

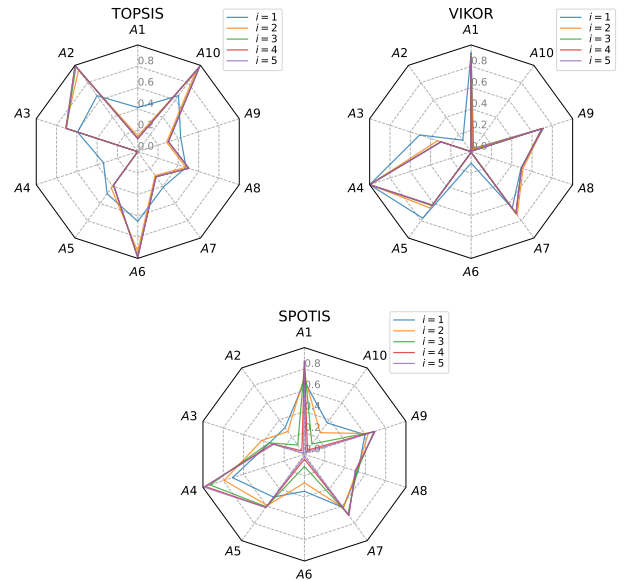


Fig. 1: Obtained preferences for TOPSIS, VIKOR, and SPOTIS methods in subsequent iterations of the proposed approach.

Among the considered methods, the SPOTIS approach had the slowest stabilization of ratings. Ranges of obtained assessments were increasing much slower than in the case of TOPSIS and VIKOR. In addition, the intervals obtained in the last iteration did not reach the limits of the SPOTIS method domain. For the first iteration, the score interval $[0.30921, 0.70877]$ was achieved, while the score interval $[0.01359, 0.99223]$ was achieved for the last iteration. Regarding the

scatter of values, it is similar to the TOPSIS method. In the first iteration, the standard deviation for the scores was 0.14258, while in the last iteration, it was 0.34732.

Using the Figure 3 the obtained rankings for each iteration are shown, where 3a shows the obtained ranking for the TOPSIS method, 3b shows the obtained ranking for the VIKOR method, and 3c shows the obtained ranking for the SPOTIS method.

The TOPSIS method, besides alternative A_4 , also obtained invariant positions in all iterations for two alternatives, i.e., alternative A_3 and alternative A_9 , which obtained fourth and seventh ranking positions, respectively. The only noticeable changes in the ranking positions for this method occurred between iteration one and iteration two.

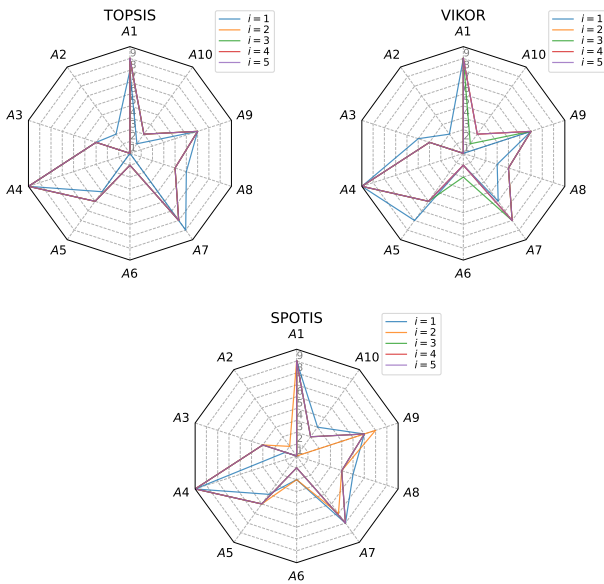


Fig. 3: Obtained rankings for TOPSIS, VIKOR, and SPOTIS methods in subsequent iterations of the proposed approach.

In the case of the VIKOR method, besides alternative A_4 , invariability of position for each iteration is also obtained by alternative A_1 . The alternative A_{10} received the most changes in its ranking position during all iterations, obtaining the first ranking position in the first iteration, obtaining the third-ranking position in the second, fourth, and fifth iterations, and obtaining the second-ranking position in the third iteration.

For the SPOTIS method, in addition to alternative A_4 , alternative A_1 also received an unchanged ranking position of 9. The most significant apparent change in ranking position for this method occurred for alternative A_{10} . In the first iteration, it achieved the fourth-ranking position. The second iteration achieved the first ranking position, and in the rest of the iterations, it achieved the third-ranking position.

A comparison of the rankings from the first and last iteration for the considered TOPSIS, VIKOR, and SPOTIS methods has been shown with the help of Figure 2. Referring to the TOPSIS method, only three alternatives, i.e., A_3 , A_4 , and A_9 are in

the same positions in the first and last rankings. The biggest difference is seen for alternative A_2 , where in the case of the first iteration, it reached the third-ranking position, while in the last iteration, it reached the first ranking position. In the case of the method, the same positions in both considered rankings were achieved by four alternatives A_1 , A_4 , A_6 , and A_9 . Only two alternatives differed by one ranking position. In contrast, four alternatives differed by two ranking positions. The SPOTIS method has the same ranking positions from the last iteration relative to the first iteration. Alternatives A_1 , A_2 , A_4 , A_7 , and A_9 remained in their ranking positions. Only one alternative differed by two positions among the rankings considered.

The Spearman's weighted correlation coefficient values for the individual rankings from iterations $i-1$ and i are presented using the table II. The TOPSIS method achieved the fastest, equally similar rankings, where as early as the third iteration, the value of r_w was 1.0. However, the VIKOR method only achieved this value in the fifth iteration, while the SPOTIS method achieved it in the fourth iteration. The most significant ranking differences are seen in the SPOTIS method, where in the first iteration, the value of r_w between the rankings was 0.85013, and in the second iteration, it was 0.93278.

TABLE II: Spearman weighted coefficient r_w values for $Rank_{i-1}$ and $Rank_i$.

Iteration	Methods		
	TOPSIS	VIKOR	SPOTIS
$i = 2$	0.92286	0.87107	0.85013
$i = 3$	1.0	0.98126	0.93278
$i = 4$	1.0	0.98126	1.0
$i = 5$	1.0	1.0	1.0

The values of the ranking similarity coefficient for the individual rankings from iterations $i-1$ and i are shown using the table III. The differences between the methods show a higher number of iterations in reaching the upper bound value of this coefficient. For example, the TOPSIS method reached the value of 1.0 in 3 iterations, the VIKOR method in 5 iterations, and the SPOTIS method in 4 iterations. Interestingly, the SPOTIS method obtained a lower value of the WS coefficient in iteration 3 than the value of the WS coefficient in iteration 2.

TABLE III: Rank similarity coefficient values WS for $Rank_{i-1}$ and $Rank_i$.

Iteration	Methods		
	TOPSIS	VIKOR	SPOTIS
$i = 2$	0.86730	0.82914	0.83945
$i = 3$	1.0	0.95089	0.83792
$i = 4$	1.0	0.95089	1.0
$i = 5$	1.0	1.0	1.0

IV. CONCLUSION

One of the essential issues of considering multiple MCDM methods evaluating the same set of decision alternatives is

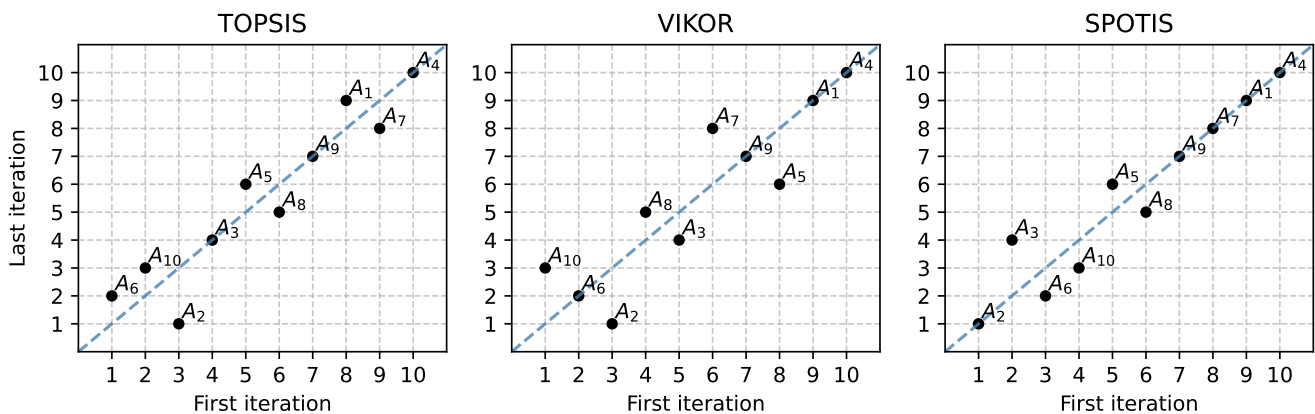


Fig. 2: Visualization of ranking similarity from first iteration and last iteration for TOPSIS, VIKOR and SPOTIS approaches.

the compromise. This paper presents a new iterative approach related to the output values of TOPSIS, VIKOR, and SPOTIS methods to create successive decision matrices. The proposed approach is very flexible and guarantees the reliability of the results, as shown by the tests performed. All considered methods obtained a compromise ranking in a minimal number of iterations. Additionally, each of the considered methods in the last iteration obtained the same ranking as the other methods. High reliability of the obtained results is also guaranteed by the used similarity coefficients of the rankings r_w and WS , which reached the upper values of their ranges.

In future research, it would be helpful to consider the effect of the number of criteria on the number of iterations needed to reach a compromise for the methods. Additionally, it would be helpful to investigate the effect of different weight allocation methods on the final values of the proposed approach. Finally, to adapt the approach in future research, it would need to be verified with the reference ranking. Future research would also need to include a broader validation of the proposed approach with more MCDM methods. In addition, an aspect related to the impact of the number of alternatives and criteria on the final results would need to be addressed.

ACKNOWLEDGMENT

The work was supported by the National Science Centre, Decision number 2021/41/B/HS4/01296 (B.K., A.S. and W.S).

REFERENCES

- [1] A. Karczmarczyk, J. Wątróbski, J. Jankowski, and E. Ziemia, "Comparative study of ict and sis measurement in polish households using a mcda-based approach," *Procedia Computer Science*, vol. 159, pp. 2616–2628, 2019.
- [2] W. Sałabun, J. Wątróbski, and A. Shekhovtsov, "Are mcda methods benchmarkable? a comparative study of topsis, vikor, copras, and promethee ii methods," *Symmetry*, vol. 12, no. 9, p. 1549, 2020.
- [3] A. M. Ghaleb, H. Kaid, A. Alsamhan, S. H. Mian, and L. Hidri, "Assessment and comparison of various mcda approaches in the selection of manufacturing process," *Advances in Materials Science and Engineering*, vol. 2020, 2020.
- [4] M. Baydaş and O. E. Elma, "An objective criteria proposal for the comparison of mcda and weighting methods in financial performance measurement: An application in borsa istanbul," *Decision Making: Applications in Management and Engineering*, vol. 4, no. 2, pp. 257–279, 2021.
- [5] Z. Liao, H. Liao, and B. Lev, "Compromise solutions for stochastic multicriteria acceptability analysis with uncertain preferences and non-monotonic criteria," *International Transactions in Operational Research*, 2021.
- [6] Ž. Stević, D. Pamučar, A. Puška, and P. Chatterjee, "Sustainable supplier selection in healthcare industries using a new mcda method: Measurement of alternatives and ranking according to compromise solution (marcos)," *Computers & Industrial Engineering*, vol. 140, p. 106231, 2020.
- [7] F. Ecer, "A consolidated mcda framework for performance assessment of battery electric vehicles based on ranking strategies," *Renewable and Sustainable Energy Reviews*, vol. 143, p. 110916, 2021.
- [8] A. Bączkiewicz, B. Kizielewicz, A. Shekhovtsov, J. Wątróbski, and W. Sałabun, "Methodical aspects of mcda based e-commerce recommender system," *Journal of Theoretical and Applied Electronic Commerce Research*, vol. 16, no. 6, pp. 2192–2229, 2021.
- [9] W. Serrai, A. Abdelli, L. Mokdad, and Y. Hammal, "Towards an efficient and a more accurate web service selection using mcda methods," *Journal of computational science*, vol. 22, pp. 253–267, 2017.
- [10] H. Zhao, B. Li, H. Lu, X. Wang, H. Li, S. Guo, W. Xue, and Y. Wang, "Economy-environment-energy performance evaluation of cchp micro-grid system: A hybrid multi-criteria decision-making method," *Energy*, vol. 240, p. 122830, 2022.
- [11] S. Chakraborty, "Topsis and modified topsis: A comparative analysis," *Decision Analytics Journal*, vol. 2, p. 100021, 2022.
- [12] P. Ziemia, A. Becker, and J. Becker, "A consensus measure of expert judgment in the fuzzy topsis method," *Symmetry*, vol. 12, no. 2, p. 204, 2020.
- [13] D. Abdul, J. Wenqi, and A. Tanveer, "Prioritization of renewable energy source for electricity generation through ahp-vikor integrated methodology," *Renewable Energy*, vol. 184, pp. 1018–1032, 2022.
- [14] J. Dezert, A. Tchamova, D. Han, and J.-M. Tacnet, "The spotis rank reversal free method for multi-criteria decision-making support," in *2020 IEEE 23rd International Conference on Information Fusion (FUSION)*. IEEE, 2020, pp. 1–8.
- [15] P. A. Zuidema, F. Babst, P. Groenendijk, V. Trouet, A. Abiyu, R. Acuña-Soto, E. Adenesky-Filho, R. Alfaro-Sánchez, J. R. V. Aragão, G. Assis-Pereira et al., "Tropical tree growth driven by dry-season climate variability," *Nature Geoscience*, pp. 1–8, 2022.
- [16] W. Sałabun and K. Urbaniak, "A new coefficient of rankings similarity in decision-making problems," in *International Conference on Computational Science*. Springer, 2020, pp. 632–645.

Towards the identification of MARCOS models based on intuitionistic fuzzy score functions

Bartłomiej Kizielewicz*, Bartosz Paradowski, Jakub Więckowski, Wojciech Sałabun

Research Team on Intelligent Decision Support Systems,
Department of Artificial Intelligence Methods and Applied Mathematics,
Faculty of Computer Science and Information Technology
West Pomeranian University of Technology in Szczecin
ul. Żołnierska 49, 71-210 Szczecin, Poland

Email: {bartlomiej-kizielewicz, jakub-wieckowski, bartosz-paradowski, wojciech.salabun}@zut.edu.pl

Abstract—We encounter uncertainty in many areas. In decision-making, it is an aspect that allows for better modeling of real-world problems. However, many methods rely on crisp numbers in their calculations. It makes it necessary to use techniques that perform this conversion. In this paper, we address the problem of score functions assessment regarding their effectiveness and usefulness in the decision-making field. The selected methods were used to convert the intuitionistic fuzzy set matrix into crisp data, then used in the multi-criteria assessment. Managing the theoretical problem showed that the used techniques provide high similarity values. Moreover, they proved to be helpful when dealing with intuitionistic fuzzy sets in the decision-making area.

I. INTRODUCTION

Many multi-criteria decision-making problems are considered in areas where data are represented using crisp numbers [1]. However, uncertainty problems are difficult to represent using this approach. Therefore, many tools based on classical arithmetic methods have been developed to model uncertainty in decision problems [2]. Such tools allow us to model real-world problems more accurately and reflect uncertain knowledge flexibly. Uncertainty modeling tools are often used in multi-criteria decision-making problems due to their high reliability [3].

Several popular tools can be used to represent uncertain knowledge. Among the classical approaches are fuzzy sets (FS), based on the idea related to partial membership [4]. Over the years, fuzzy sets have seen many developments: Hesitant Fuzzy Sets (HFS) [5], Fermatean Fuzzy Sets (FFS) [6], Picture Fuzzy Sets (PFS) [7], or Intuitionistic Fuzzy Sets (IFS) [8]. Indeed, the main advantage of the generalization of fuzzy sets is a new approach to uncertainty modeling that considers new degrees of membership, which gives the expert the ability to adapt to the characteristics of the problem [9].

One of the most popular tools based on the idea of fuzzy sets is Intuitionistic Fuzzy Sets. This tool introduces the possibility of determining the degree of membership and non-membership, thanks to which it is helpful in many areas such as decision-making and medical diagnosis [10], [11]. The wide use of Intuitionistic Fuzzy Sets has led to the development of this approach. A new similarity measure between intuitionistic fuzzy sets was proposed by Gohain et al. [12]. Szmids et al.

proposed a new proposal for attribute selection in models expressed by intuitionistic fuzzy sets [13]. Thao proposed new divergence measures of intuitionistic fuzzy sets from Archimedean t -conorm operators [14].

Using an extension of multi-criteria decision-making methods with fuzzy logic makes it possible to change the problem environment from crisp to uncertain. However, most Multi-Criteria Decision-Making (MCDM) related approaches operate in an environment based on crisp numbers [15]. To convert fuzzy data to crisp data, one can use point functions, whose idea in multi-criteria decision making was originally proposed by Chen and Tan [16]. However, the existence of multiple scoring functions means that their use within the same problem may be characterized by obtaining different results [17]. It creates a research gap that needs to be filled and determines which score function to select so that the results are satisfactory.

In this paper, we used five different score functions to convert Intuitionistic Fuzzy Sets to crisp values and assess the obtained decision matrix with the Measurement Alternatives and Ranking according to COmpromise Solution (MARCOS) method. The simulated data was used as the inputs to show the performance of the presented approach in the theoretical problem. Obtained results were then compared with selected correlation coefficients to point out the similarity of the used paths. The purpose of the study is to indicate the influence of the used score function regarding the differences obtained in multi-criteria ranking.

The rest of the paper is organized as follows. Section 2 presents the preliminaries of the IFS, the scores functions, the MARCOS method and selected similarity coefficients. In Section 3, the study case is shown, where the theoretical problem of the functioning of the different scores function is raised. Section 4 includes the description of the results obtained from the examined research. Finally, in Section 5, the summary is presented, and the conclusions are drawn.

II. PRELIMINARIES

A. Intuitionistic Fuzzy Sets

Definition II.1. An IFS A in a universe X is defined as an object of the following form:

$$A = \{ \langle x_j, \mu_j, \nu_j \rangle \mid x_j \in X \} \tag{1}$$

where $\mu : X \rightarrow [0, 1]$ and $\nu : X \rightarrow [0, 1]$ such that $0 \leq \mu_j + \nu_j \leq 1$ for all $x_j \in X$. The values of μ_j and ν_j represent the degrees of membership and non-membership of $x_j \in X$ in A respectively [17].

For every $A \in IFS(X)$ (the class of IFSs in the universe X), the value of

$$\pi_j = 1 - \mu_j - \nu_j \tag{2}$$

represents the degree of hesitation (or uncertainty) associated with the membership of element $x_j \in X$ in IFS A , where $0 \leq \pi_j \leq 1$.

B. Score Functions

The purpose of the score function is to convert the uncertain data representation to a crisp value. Different approaches to performing such an action obtain diverse values as a final output. Selected score functions and the formulas for their calculations are presented below [17], [18], [19].

$$S_I(X_{ij}) = \mu_{ij} - \nu_{ij} \tag{3}$$

$$S_{II}(X_{ij}) = \mu_{ij} - \nu_{ij} \cdot \pi_{ij} \tag{4}$$

$$S_{III}(X_{ij}) = \mu_{ij} - \left(\frac{\nu_{ij} + \pi_{ij}}{2} \right) \tag{5}$$

$$S_{IV}(X_{ij}) = \left(\frac{\mu_{ij} + \nu_{ij}}{2} \right) - \pi_{ij} \tag{6}$$

$$S_V(X_{ij}) = \gamma \cdot \mu_{ij} + (1 - \gamma) \cdot (1 - \nu_{ij}), \quad \gamma \in [0, 1] \tag{7}$$

where $S_I(X_{ij}), S_{II}(X_{ij}), S_V(X_{ij}) \in [-1, 1]$, $S_{III}(X_{ij}) \in [-0.5, 1]$, and $S_{IV}(X_{ij}) \in [-1, 0.5]$.

C. MARCOS method

The Measurement Alternatives and Ranking according to COMpromise Solution (MARCOS) method was introduced by Željko Stević in 2020 [20] as new multi-criteria decision making method, which was presented on study case of sustainable supplier selection in healthcare industries. This method provides new approach to solve decision problems, as it considers an anti-ideal and ideal solution at the initial steps of consideration of the problem. Moreover it proposes new way to determine utility functions and their further aggregation, while maintaining stability in the problems requiring large set of alternatives and criteria.

Step 1. The initial step requires to define set of n criteria and m alternatives to create decision matrix.

Step 2. Next, the extended initial matrix X should be formed by defining ideal (AI) and anti-ideal(AAI) solution.

$$X = \begin{matrix} & \begin{matrix} A_{I1} \\ A_1 \\ A_2 \\ \dots \\ A_m \\ A_I \end{matrix} \\ \begin{matrix} A_{II} \\ A_1 \\ A_2 \\ \dots \\ A_m \\ A_I \end{matrix} & \begin{bmatrix} x_{aa1} & x_{aa2} & \dots & x_{aan} \\ x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \dots & \dots & \dots & \dots \\ x_{m1} & x_{m2} & \dots & x_{mn} \\ x_{ai1} & x_{ai2} & \dots & x_{ain} \end{bmatrix} \end{matrix} \tag{8}$$

The anti-ideal solution (AAI) which is the worst alternative is defined by equation (9), whereas the ideal solution (AI) is the best alternative in the problem at hand defined by equation (10).

$$AAI = \min_i x_{ij} \quad \text{if } j \in B \text{ and } \max_i x_{ij} \quad \text{if } j \in C \tag{9}$$

$$AI = \max_i x_{ij} \quad \text{if } j \in B \text{ and } \min_i x_{ij} \quad \text{if } j \in C \tag{10}$$

where B is a benefit group of criteria and C is a group of cost criteria.

Step 3. After defining anti-ideal and ideal solutions, the extended initial matrix X needs to be normalized, by applying equations (11) and (12) creating normalized matrix N .

$$n_{ij} = \frac{x_{ai}}{x_{ij}} \quad \text{if } j \in C \tag{11}$$

$$n_{ij} = \frac{x_{ij}}{x_{ai}} \quad \text{if } j \in B \tag{12}$$

Step 4. The weight for each criterion needs to be defined to present its importance in accordance to others. The weighted matrix V needs to be calculated by multiplying the normalized matrix N with the weight vector through equation (13).

$$v_{ij} = n_{ij} \times w_j \tag{13}$$

Step 5. Next, the utility degree K of alternatives in relation to the anti-ideal and ideal solutions needs to be calculated by using equations (14) and (15)

$$K_i^- = \frac{\sum_{i=1}^n v_{ij}}{\sum_{i=1}^n v_{ai}} \tag{14}$$

$$K_i^+ = \frac{\sum_{i=1}^n v_{ij}}{\sum_{i=1}^n v_{ai}} \tag{15}$$

Step 6. The utility function f of alternatives, which is the compromise of the observed alternative in relation to the ideal and anti-ideal solution, needs to be determined. Its done using equation (16).

$$f(K_i) = \frac{K_i^+ + K_i^-}{1 + \frac{1-f(K_i^+)}{f(K_i^+)} + \frac{1-f(K_i^-)}{f(K_i^-)}} \tag{16}$$

where $f(K_i^-)$ represents the utility function in relation to the anti-ideal solution and $f(K_i^+)$ represents the utility function in relation to the ideal solution.

Utility functions in relation to the ideal and anti-ideal solution are determined by applying equations (17) and (18).

$$f(K_i^-) = \frac{K_i^+}{K_i^+ + K_i^-} \tag{17}$$

$$f(K_i^+) = \frac{K_i^-}{K_i^+ + K_i^-} \tag{18}$$

Step 7. Finally, rank alternatives accordingly to the values of the utility functions. The higher the value the better is an alternative.

D. Rank similarity coefficients

In order to compare the performance of the score functions, it would be useful to compare the rankings obtained after evaluating the values calculated using these functions. For this purpose, one can use rank similarity coefficients, which are often used in the literature to compare the resulting rankings. In the case of our study, we decided to use weighted Spearman’s correlation coefficient, which allows comparing rankings considering alternatives rated the best as more significant, and the WS ranking similarity coefficient, which the main assumption that the positions of top of the rankings has a more significant influence on similarity. The formulas for calculation of both coefficients are presented below in equation (19) for weighted Spearman’s correlation and equation (20) for WS rank similarity coefficient.

$$r_w = 1 - \frac{6 \cdot \sum_{i=1}^n (x_i - y_i)^2 ((N - x_i + 1) + (N - y_i + 1))}{n \cdot (n^3 + n^2 - n - 1)} \tag{19}$$

$$WS = 1 - \sum_{i=1}^n \left(2^{-x_i} \frac{|x_i - y_i|}{\max\{|x_i - 1|, |x_i - N|\}} \right) \tag{20}$$

III. STUDY CASE

The use of fuzzy sets in multi-criteria problems is a popular approach to solving problems where uncertainty arises. It allows greater flexibility in modeling input data, thus ensuring that the actual values that determine the parameters can be represented. However, in many cases, the criteria are not considered in a binary way, or the corresponding values are not known precisely. Fuzzy sets are one of the possible ways to represent uncertainty [21]. In the following, we focus our attention on the problem of using Intuitionistic Fuzzy Sets and different score functions to point out differences and similarities in the results obtained by using these tools.

A randomly generated decision matrix of 6 alternatives and 4 criteria was used in the study. Each matrix element is represented in the form of an IFS, where the first value indicates the value of decisiveness, while the second determines the degree of indecisiveness. Then, based on the score functions described above, conversions of the uncertain matrix to a matrices represented in the form of sharp numbers were performed. The generated matrix is shown in Table I.

The purpose of this operation is the need to indicate how a given score function affects the process of converting the data to a crisp form. Furthermore, it is crucial to determine whether the obtained matrices influence the obtained result through a multi-criteria analysis.

IV. RESULTS

A. Small example

Each type of previously presented score function was used to calculate crisp values for the matrix, which were shown in Tables respective to the used function. Table II presents values obtained by use of score function S_I . In the case of this score function, the spread of values in the range $[-1, 1]$ is around 1.69, which might mean that this specific score function differentiates well between alternative values.

TABLE II
CRISP SMALL DECISION MATRIX CALCULATED WITH S_I SCORE FUNCTION.

A_i	C_1	C_2	C_3	C_4
A_1	-0.039081	0.137740	-0.351916	0.691538
A_2	-0.558417	-0.455956	-0.649449	0.010564
A_3	-0.405613	-0.006527	0.361583	0.244181
A_4	-0.211142	-0.122171	-0.596428	0.159583
A_5	0.479454	-0.860203	0.830694	-0.110298
A_6	0.375293	0.328710	-0.797545	-0.822696

In Table III values calculated through execution of score function S_{II} are presented. This function is defined as the degree of membership minus the product of the non-membership and hesitation degrees, and even though it provides values from the same range as S_I , it can be seen that there are less negative values. Moreover, it is clear that in this example, the spread of calculated values is significantly smaller, as in this case, it’s around 0.99.

TABLE III
CRISP SMALL DECISION MATRIX CALCULATED WITH S_{II} SCORE FUNCTION.

A_i	C_1	C_2	C_3	C_4
A_1	0.041174	0.510925	0.243452	0.726737
A_2	0.205952	0.075299	0.021391	0.089091
A_3	-0.041371	0.060761	0.568287	0.587950
A_4	-0.141754	0.392034	0.001438	0.324662
A_5	0.506242	-0.056156	0.849442	0.389249
A_6	0.383334	0.368876	-0.004356	-0.014798

The values obtained through the equation of score function S_{III} are shown in Table IV. This specific function operates in the range $[-0.5, 1]$ and is similar to the previous one but subtracts the arithmetic mean of the non-membership and hesitation degrees. As a result, provided values spread around 1.24, which translates into a high differentiation of the individual IFS values from the initial decision matrix.

TABLE I
SMALL DECISION MATRIX REPRESENTED BY INTUITIONISTIC FUZZY SETS.

A_i	C_1 (μ, ν)	C_2 (μ, ν)	C_3 (μ, ν)	C_4 (μ, ν)
A_1	(0.17125,0.21033)	(0.53664,0.39890)	(0.28872,0.64063)	(0.73657,0.04503)
A_2	(0.21496,0.77338)	(0.18588,0.64183)	(0.11440,0.76385)	(0.20609,0.19553)
A_3	(0.13443,0.54004)	(0.17854,0.18506)	(0.60514,0.24356)	(0.60220,0.35801)
A_4	(0.03524,0.24638)	(0.41634,0.53851)	(0.11939,0.71582)	(0.40974,0.25016)
A_5	(0.52621,0.04675)	(0.02438,0.88458)	(0.85215,0.02146)	(0.41781,0.52811)
A_6	(0.39471,0.01942)	(0.41035,0.08164)	(0.06241,0.85995)	(0.05100,0.87369)

TABLE IV
CRISP SMALL DECISION MATRIX CALCULATED WITH S_{II} SCORE FUNCTION.

A_i	C_1	C_2	C_3	C_4
A_1	-0.243131	0.304959	-0.066927	0.604858
A_2	-0.177553	-0.221182	-0.328406	-0.190864
A_3	-0.298357	-0.232197	0.407707	0.403293
A_4	-0.447136	0.124515	-0.320909	0.114611
A_5	0.289310	-0.463433	0.778231	0.126715
A_6	0.092065	0.115525	-0.406387	-0.423503

TABLE VI
CRISP SMALL DECISION MATRIX CALCULATED WITH S_V SCORE FUNCTION.

A_i	C_1	C_2	C_3	C_4
A_1	0.480460	0.568870	0.324042	0.845769
A_2	0.220791	0.272022	0.175275	0.505282
A_3	0.297193	0.496736	0.680791	0.622091
A_4	0.394429	0.438915	0.201786	0.579791
A_5	0.739727	0.069898	0.915347	0.444851
A_6	0.687646	0.664355	0.101228	0.088652

The function S_{IV} is defined as the arithmetic mean of the membership and non-membership degrees minus the hesitation degree, which operates in the range $[-1, 0.5]$. The spread of the values obtained is around 1.06, which is slightly less than the previous function, but still shows that the IFS values are significantly differentiated from each other.

TABLE V
CRISP SMALL DECISION MATRIX CALCULATED WITH S_{IV} SCORE FUNCTION.

A_i	C_1	C_2	C_3	C_4
A_1	-0.427641	0.403307	0.394019	0.172409
A_2	0.482520	0.241570	0.317362	-0.397573
A_3	0.011706	-0.454602	0.273041	0.440315
A_4	-0.577559	0.432286	0.252824	-0.010152
A_5	-0.140560	0.363438	0.310422	0.418877
A_6	-0.378809	-0.262015	0.383544	0.387039

Values for last score function, namely S_V are presented in Table VI. This function represents a mixed result of positive and negative outcome expectations and operates in the same range as S_I and S_{II} . In this case, no negative values were received even though the range in which operates this function includes negative values. The spread of values received from this function is around 0.85, which is the lowest of presented score functions, considering its range.

Table VII presents preference values calculated by execution of MARCOS method. The values of preference for respective alternatives show the differences between considered score functions. The function S_I has irregular distribution, where only one value is significantly higher than the rest. But in the case of this function, the difference between the highest and lowest value is almost 0.3, which shows that the values do not have a high spread. On the contrary, the score function S_{II} provides a higher spread of 0.426, which might be preferable as it better distinguishes the differences between alternatives.

Score function S_{III} provides the smallest spread of values of all functions, namely 0.047. In such a case, it may be perceived as the difference between alternatives is insignificant, which is rarely preferable in case of decision problems. Evaluated values from score function S_{IV} yielded values that spread around 0.23, which is not the highest of presented score functions but might be useful in some cases. The last score function S_V provided the highest values in this Table, which might be visually better perceived by some decision-makers, as the differences between the alternatives are more readily apparent. The spread is around 0.33, which is the second-highest. Considering those values, functions S_{II} and S_V are the most representative and might be preferred by numerous decision-makers.

TABLE VII
PREFERENCES FOR SMALL DECISION MATRIX COMPUTED WITH MARCOS METHOD FOR S_I - S_V SCORE FUNCTIONS.

A_i	S_I	S_{II}	S_{III}	S_{IV}	S_V
A_1	-0.064287	0.584944	-0.017914	0.174813	0.700419
A_2	0.233428	0.173407	0.034649	0.177619	0.366605
A_3	0.005450	0.374305	0.010045	0.080901	0.643830
A_4	0.091397	0.231148	0.022694	0.051248	0.514291
A_5	0.054616	0.599214	-0.011511	0.278166	0.644428
A_6	0.025817	0.359104	0.008765	0.056061	0.525352

Figure 1 presents alternatives ranked by preference obtained through considered score functions. On the graph, the differences in evaluation are clearly visible as, for example, the score function S_{III} and S_I ranked alternative A_1 as the worst. In contrast, score function S_V ranked this alternative as the worst. On the other hand, almost all functions placed alternative A_6 fourth.

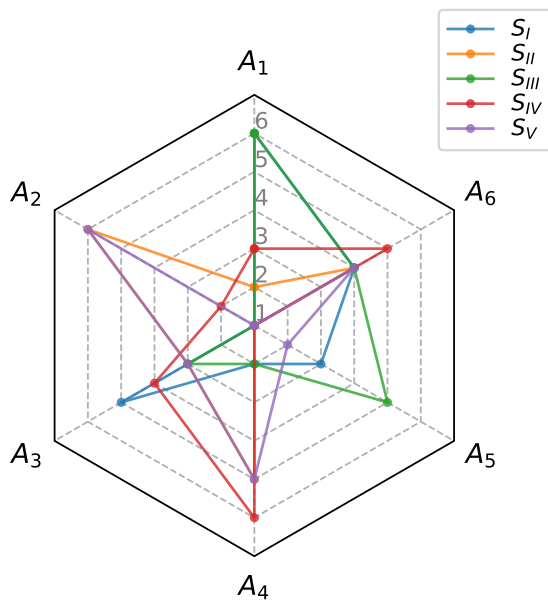


Fig. 1. Radar chart of MARCOS rankings.

To better visualize differences between presented score functions, rankings obtained by execution of the MARCOS method were compared using similarity coefficients. The first coefficient, namely Weighted Spearman’s correlation coefficient, is presented in Figure 2 as a correlation matrix in the form of a heatmap. This coefficient shows high similarities of rankings, which resulted through execution of score functions S_{II} and S_V . The previous examination showed that those two functions behave rather similarly, resulting in crisp values of IFS. The next pair of functions that are quite similar is S_{III} and S_I .

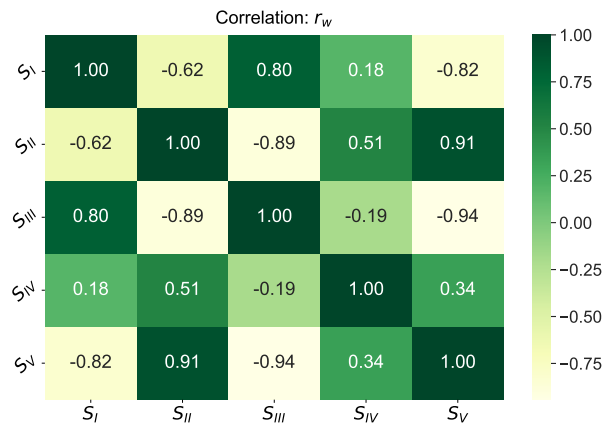


Fig. 2. Weighted Spearman’s correlation heatmap of MARCOS rankings for small decision matrix.

Additionally, the rankings were compared using the WS rank similarity coefficient, which as well is presented as a correlation matrix in the form of a heatmap as Figure 3. This coefficient shows which pair of compared rankings are not symmetrical, meaning that rankings are not identical neither the change in position is between exactly the same alternatives. As it can be seen once again, the pair S_I and S_{III} and pair S_{II} and S_V are characterized by a high degree of similarity. Moreover, comparing S_{II} and S_{IV} where S_{II} is treated as yields high similarity.

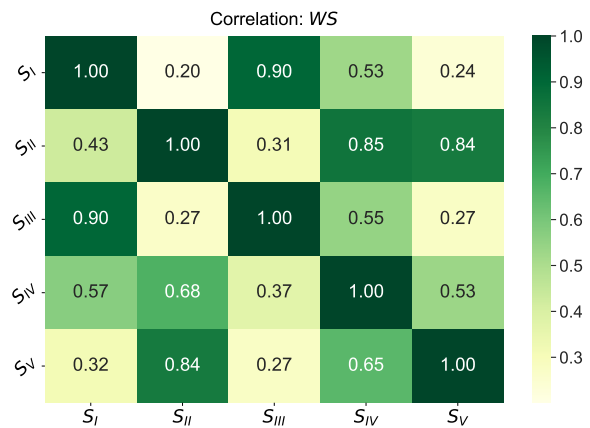


Fig. 3. WS correlation heatmap of MARCOS rankings for small decision matrix.

B. Big example

The next example that was taken into consideration consists of twenty alternatives and six criteria, which created the decision matrix presented in Table VIII. This approach provides a view of how specific score functions behave in larger multi-criteria decision problems.

Similarly to the smaller example, for a decision matrix consisting of IFS, crisp matrix was calculated using score

TABLE VIII
BIG DECISION MATRIX REPRESENTED BY INTUITIONISTIC FUZZY SETS.

A_i	C_1 (μ, ν)	C_2 (μ, ν)	C_3 (μ, ν)	C_4 (μ, ν)	C_5 (μ, ν)	C_6 (μ, ν)
A_1	(0.09095,0.71850)	(0.13774,0.49164)	(0.20716,0.00628)	(0.06600,0.90771)	(0.93253,0.01122)	(0.17497,0.50594)
A_2	(0.19634,0.76889)	(0.05385,0.74241)	(0.07135,0.15414)	(0.44855,0.19833)	(0.15521,0.30392)	(0.41031,0.09120)
A_3	(0.02148,0.09223)	(0.37886,0.44763)	(0.20687,0.35686)	(0.11724,0.55262)	(0.15676,0.80864)	(0.28854,0.00747)
A_4	(0.80399,0.04354)	(0.10795,0.17483)	(0.80161,0.11037)	(0.49469,0.30475)	(0.18160,0.29511)	(0.02199,0.37757)
A_5	(0.59997,0.06593)	(0.52386,0.23834)	(0.25322,0.03477)	(0.86408,0.08472)	(0.14660,0.03157)	(0.35019,0.03213)
A_6	(0.17915,0.01657)	(0.35069,0.17044)	(0.23634,0.71446)	(0.71924,0.26769)	(0.16976,0.73265)	(0.13227,0.62576)
A_7	(0.39394,0.41539)	(0.59632,0.02533)	(0.28756,0.53541)	(0.03969,0.82749)	(0.44033,0.44300)	(0.18513,0.47813)
A_8	(0.09366,0.55802)	(0.78447,0.18155)	(0.15418,0.51586)	(0.51218,0.19691)	(0.54950,0.29227)	(0.78470,0.19383)
A_9	(0.20754,0.11127)	(0.02822,0.77145)	(0.11259,0.43723)	(0.10478,0.83100)	(0.43970,0.01513)	(0.36435,0.54747)
A_{10}	(0.53145,0.38788)	(0.51920,0.27552)	(0.47281,0.39902)	(0.29417,0.18583)	(0.44656,0.32535)	(0.56993,0.28041)
A_{11}	(0.41454,0.55076)	(0.60336,0.17182)	(0.25771,0.20216)	(0.43993,0.40186)	(0.34019,0.46780)	(0.28601,0.21097)
A_{12}	(0.00178,0.27447)	(0.21928,0.08916)	(0.22282,0.15183)	(0.52120,0.09153)	(0.10013,0.27936)	(0.44117,0.20162)
A_{13}	(0.57693,0.40495)	(0.07804,0.58253)	(0.54650,0.08093)	(0.67845,0.08846)	(0.19737,0.36442)	(0.72426,0.00283)
A_{14}	(0.03320,0.27491)	(0.17390,0.71133)	(0.78701,0.18097)	(0.15871,0.82272)	(0.68930,0.04541)	(0.16032,0.47469)
A_{15}	(0.14737,0.32855)	(0.62481,0.01155)	(0.34158,0.62958)	(0.61442,0.01856)	(0.85899,0.08471)	(0.29505,0.22883)
A_{16}	(0.46437,0.49039)	(0.46334,0.19572)	(0.20792,0.70879)	(0.08940,0.36645)	(0.00819,0.58278)	(0.04862,0.11211)
A_{17}	(0.17621,0.05731)	(0.39896,0.25935)	(0.54071,0.45444)	(0.11264,0.77514)	(0.70301,0.02073)	(0.40974,0.52313)
A_{18}	(0.03117,0.46166)	(0.48374,0.35310)	(0.09871,0.32470)	(0.15036,0.08845)	(0.25587,0.41844)	(0.03895,0.57066)
A_{19}	(0.37151,0.06187)	(0.29695,0.33688)	(0.03437,0.25546)	(0.63877,0.03488)	(0.87320,0.08054)	(0.37052,0.44601)
A_{20}	(0.26295,0.58775)	(0.32911,0.29911)	(0.08203,0.25447)	(0.74980,0.20059)	(0.26421,0.04034)	(0.46111,0.50190)

functions. The resultant matrix with crisp values is presented in Table IX.

TABLE IX
PREFERENCES FOR BIG DECISION MATRIX COMPUTED WITH MARCOS METHOD FOR S_I - S_V SCORE FUNCTIONS.

A_i	S_I	S_{II}	S_{III}	S_{IV}	S_V
A_1	-0.036549	0.190882	-0.085154	-0.041990	0.451390
A_2	-0.030858	0.146254	-0.109542	0.059565	0.465973
A_3	-0.028761	0.126735	-0.129638	0.081650	0.459555
A_4	0.033628	0.355654	0.068392	0.049577	0.658271
A_5	0.067533	0.500764	0.122542	0.119319	0.763359
A_6	-0.019921	0.272210	-0.038713	-0.046104	0.489817
A_7	-0.019389	0.278088	-0.007072	-0.101019	0.487457
A_8	0.029940	0.462223	0.143382	-0.127860	0.643773
A_9	-0.050486	0.130363	-0.126984	0.004657	0.411298
A_{10}	0.030223	0.483942	0.140541	-0.105493	0.646673
A_{11}	0.013713	0.371025	0.056929	-0.055130	0.590749
A_{12}	0.014285	0.170247	-0.078569	0.216041	0.596157
A_{13}	0.036599	0.465783	0.134680	-0.048441	0.671756
A_{14}	-0.020070	0.282331	-0.011157	-0.077203	0.501384
A_{15}	0.045216	0.480237	0.126290	-0.027890	0.697439
A_{16}	-0.030772	0.129525	-0.105932	0.042165	0.452381
A_{17}	0.005362	0.395817	0.049615	-0.063564	0.574704
A_{18}	-0.033528	0.026981	-0.153698	0.105713	0.449290
A_{19}	0.034387	0.394112	0.073114	0.025449	0.672488
A_{20}	0.005835	0.327543	0.020393	-0.003544	0.576575

The first function, namely S_I yielded results presented in Table X. In this case, the standard deviation is 0.411, which is pretty high considering the range of this function, and it tells us that this specific function provided differentiated results. Moreover, the spread of those values is 1.76, which once again, as in the smaller numerical example, shows that this function makes use of a significant part of the range it operates in.

TABLE X
CRISP BIG DECISION MATRIX CALCULATED WITH S_I SCORE FUNCTION.

A_i	C_1	C_2	C_3	C_4	C_5	C_6
A_1	-0.6276	-0.3539	0.2009	-0.8417	0.9213	-0.3310
A_2	-0.5725	-0.6886	-0.0828	0.2502	-0.1487	0.3191
A_3	-0.0708	-0.0688	-0.1500	-0.4354	-0.6519	0.2811
A_4	0.7605	-0.0669	0.6912	0.1899	-0.1135	-0.3556
A_5	0.5340	0.2855	0.2184	0.7794	0.1150	0.3181
A_6	0.1626	0.1802	-0.4781	0.4516	-0.5629	-0.4935
A_7	-0.0215	0.5710	-0.2478	-0.7878	-0.0027	-0.2930
A_8	-0.4644	0.6029	-0.3617	0.3153	0.2572	0.5909
A_9	0.0963	-0.7432	-0.3246	-0.7262	0.4246	-0.1831
A_{10}	0.1436	0.2437	0.0738	0.1083	0.1212	0.2895
A_{11}	-0.1362	0.4315	0.0555	0.0381	-0.1276	0.0750
A_{12}	-0.2727	0.1301	0.0710	0.4297	-0.1792	0.2395
A_{13}	0.1720	-0.5045	0.4656	0.5900	-0.1670	0.7214
A_{14}	-0.2417	-0.5374	0.6060	-0.6640	0.6439	-0.3144
A_{15}	-0.1812	0.6133	-0.2880	0.5959	0.7743	0.0662
A_{16}	-0.0260	0.2676	-0.5009	-0.2770	-0.5746	-0.0635
A_{17}	0.1189	0.1396	0.0863	-0.6625	0.6823	-0.1134
A_{18}	-0.4305	0.1306	-0.2260	0.0619	-0.1626	-0.5317
A_{19}	0.3096	-0.0399	-0.2211	0.6039	0.7927	-0.0755
A_{20}	-0.3248	0.0300	-0.1724	0.5492	0.2239	-0.0408

The results obtained using the S_{II} function are presented in the Table XI. In this case, values are characterized by a standard deviation of 0.29 and a spread of 1.16. Because this function operates in the same interval as S_I , namely $[-1, 1]$, they can be easily compared. And just as in the small numerical example, here too, the function S_{II} achieves smaller values of spread and standard deviation.

TABLE XI
CRISP BIG DECISION MATRIX CALCULATED WITH S_{II} SCORE FUNCTION.

A_i	C_1	C_2	C_3	C_4	C_5	C_6
A_0	-0.0460	-0.0445	0.2022	0.0421	0.9319	0.0135
A_1	0.1696	-0.0974	-0.0480	0.3785	-0.0092	0.3648
A_2	-0.0603	0.3012	0.0512	-0.0652	0.1288	0.2833
A_3	0.7974	-0.0174	0.7919	0.4336	0.0272	-0.2047
A_4	0.5779	0.4672	0.2285	0.8597	0.1207	0.3303
A_5	0.1658	0.2691	0.2012	0.7157	0.0983	-0.0191
A_6	0.3147	0.5867	0.1928	-0.0702	0.3887	0.0241
A_7	-0.1007	0.7783	-0.0160	0.4549	0.5033	0.7805
A_8	0.1317	-0.1263	-0.0842	0.0514	0.4315	0.3161
A_9	0.5002	0.4626	0.4217	0.1975	0.3724	0.5280
A_{10}	0.3954	0.5647	0.1485	0.3763	0.2504	0.1799
A_{11}	-0.1969	0.1576	0.1279	0.4858	-0.0732	0.3691
A_{12}	0.5696	-0.1197	0.5163	0.6578	0.0377	0.7235
A_{13}	-0.1570	0.0923	0.7812	0.1434	0.6773	-0.0129
A_{14}	-0.0248	0.6206	0.3234	0.6076	0.8542	0.1861
A_{15}	0.4422	0.3966	0.1489	-0.1100	-0.2302	-0.0455
A_{16}	0.1323	0.3103	0.5385	0.0256	0.6973	0.3746
A_{17}	-0.2030	0.4261	-0.0885	0.0830	0.1196	-0.1838
A_{18}	0.3365	0.1736	-0.1470	0.6274	0.8695	0.2887
A_{19}	0.1752	0.2179	-0.0868	0.7399	0.2362	0.4425

The score function S_{III} yielded values presented in Table XII. This function operates in a different range than the two previous. Considering the operative range of this function, the standard deviation value of 0.35 and spread of 1.39 are definitely high values. Results similar to those obtained in the small numerical example show that this function is stable and, at the same time, uses a large part of the interval in which it operates, providing relatively different values for the different alternatives.

TABLE XII
CRISP BIG DECISION MATRIX CALCULATED WITH S_{III} SCORE FUNCTION.

A_i	C_1	C_2	C_3	C_4	C_5	C_6
0	-0.3636	-0.2934	-0.1893	-0.4010	0.8988	-0.2375
1	-0.2055	-0.4192	-0.3930	0.1728	-0.2672	0.1155
2	-0.4678	0.0683	-0.1897	-0.3241	-0.2649	-0.0672
3	0.7060	-0.3381	0.7024	0.2420	-0.2276	-0.4670
4	0.3999	0.2858	-0.1202	0.7961	-0.2801	0.0253
5	-0.2313	0.0260	-0.1455	0.5789	-0.2454	-0.3016
6	0.0909	0.3945	-0.0687	-0.4405	0.1605	-0.2223
7	-0.3595	0.6767	-0.2687	0.2683	0.3243	0.6770
8	-0.1887	-0.4577	-0.3311	-0.3428	0.1595	0.0465
9	0.2972	0.2788	0.2092	-0.0587	0.1698	0.3549
10	0.1218	0.4050	-0.1134	0.1599	0.0103	-0.0710
11	-0.4973	-0.1711	-0.1658	0.2818	-0.3498	0.1618
12	0.3654	-0.3829	0.3197	0.5177	-0.2039	0.5864
13	-0.4502	-0.2391	0.6805	-0.2619	0.5339	-0.2595
14	-0.2789	0.4372	0.0124	0.4216	0.7885	-0.0574
15	0.1966	0.1950	-0.1881	-0.3659	-0.4877	-0.4271
16	-0.2357	0.0984	0.3111	-0.3310	0.5545	0.1146
17	-0.4532	0.2256	-0.3519	-0.2745	-0.1162	-0.4416
18	0.0573	-0.0546	-0.4484	0.4582	0.8098	0.0558
19	-0.1056	-0.0063	-0.3770	0.6247	-0.1037	0.1917

Table XIII presents results obtained using function S_{IV} . The calculated spread of values, being around 1.32, similar to the functions S_I and S_{II} shows significant use of the range in which this function operates. Moreover, the standard deviation value of about 0.36 is close to the value obtained by

the function S_{III} , which might indicate that those functions might yield similar results.

TABLE XIII
CRISP BIG DECISION MATRIX CALCULATED WITH S_{IV} SCORE FUNCTION.

A_i	C_1	C_2	C_3	C_4	C_5	C_6
0	0.2142	-0.0559	-0.6798	0.4606	0.4156	0.0214
1	0.4478	0.1944	-0.6618	-0.0297	-0.3113	-0.2477
2	-0.8294	0.2397	-0.1544	0.0048	0.4481	-0.5560
3	0.2713	-0.5758	0.3680	0.1992	-0.2849	-0.4007
4	-0.0012	0.1433	-0.5680	0.4232	-0.7327	-0.4265
5	-0.7064	-0.2183	0.4262	0.4804	0.3536	0.1370
6	0.2140	-0.0675	0.2345	0.3008	0.3250	-0.0051
7	-0.0225	0.4490	0.0051	0.0636	0.2627	0.4678
8	-0.5218	0.1995	-0.1753	0.4037	-0.3178	0.3677
9	0.3790	0.1921	0.3077	-0.2800	0.1579	0.2755
10	0.4480	0.1628	-0.3102	0.2627	0.2120	-0.2545
11	-0.5856	-0.5373	-0.4380	-0.0809	-0.4308	-0.0358
12	0.4728	-0.0092	-0.0589	0.1504	-0.1573	0.0906
13	-0.5378	0.3278	0.4520	0.4721	0.1021	-0.0475
14	-0.2861	-0.0455	0.4567	-0.0505	0.4155	-0.2142
15	0.4321	-0.0114	0.3751	-0.3162	-0.1135	-0.7589
16	-0.6497	-0.0125	0.4927	0.3317	0.0856	0.3993
17	-0.2608	0.2553	-0.3649	-0.6418	0.0115	-0.0856
18	-0.3499	-0.0493	-0.5653	0.0105	0.4306	0.2248
19	0.2761	-0.0577	-0.4952	0.4256	-0.5432	0.4445

Table XIV presents values calculated using function S_V . The standard deviation of calculated values is 0.21, whereas the spread value is 0.88. This function operates in the same range as functions S_I and S_{II} , which makes it the worst in diversifying values in comparison to those two. Even though a small standard deviation and spread characterize those values, this function might be useful when such values are expected.

TABLE XIV
CRISP BIG DECISION MATRIX CALCULATED WITH S_V SCORE FUNCTION.

A_i	C_1	C_2	C_3	C_4	C_5	C_6
0	0.1862	0.3231	0.6004	0.0791	0.9607	0.3345
1	0.2137	0.1557	0.4586	0.6251	0.4256	0.6596
2	0.4646	0.4656	0.4250	0.2823	0.1741	0.6405
3	0.8802	0.4666	0.8456	0.5950	0.4432	0.3222
4	0.7670	0.6428	0.6092	0.8897	0.5575	0.6590
5	0.5813	0.5901	0.2609	0.7258	0.2186	0.2533
6	0.4893	0.7855	0.3761	0.1061	0.4987	0.3535
7	0.2678	0.8015	0.3192	0.6576	0.6286	0.7954
8	0.5481	0.1284	0.3377	0.1369	0.7123	0.4084
9	0.5718	0.6218	0.5369	0.5542	0.5606	0.6448
10	0.4319	0.7158	0.5278	0.5190	0.4362	0.5375
11	0.3637	0.5651	0.5355	0.7148	0.4104	0.6198
12	0.5860	0.2478	0.7328	0.7950	0.4165	0.8607
13	0.3791	0.2313	0.8030	0.1680	0.8219	0.3428
14	0.4094	0.8066	0.3560	0.7979	0.8871	0.5331
15	0.4870	0.6338	0.2496	0.3615	0.2127	0.4683
16	0.5595	0.5698	0.5431	0.1687	0.8411	0.4433
17	0.2848	0.5653	0.3870	0.5310	0.4187	0.2341
18	0.6548	0.4800	0.3895	0.8019	0.8963	0.4623
19	0.3376	0.5150	0.4138	0.7746	0.6119	0.4796

The rankings obtained using the MARCOS method are grouped in the barplot shown in Figure 4. As can be seen, the obtained rankings differ significantly from each other, highlighting how important it is to choose an appropriate score

function. Additionally, it can be seen that on the podium of the ranking, the functions S_{II} , S_{III} , and S_V behave similarly. Still, in the further positions, significant discrepancies appear.



Fig. 5. Weighted Spearman's correlation heatmap of MARCOS rankings for big decision matrix.

The correlations of the rankings obtained from the big decision matrix data are shown in Figures 5 and 6 using heatmaps. The former, describing values for the weighted Spearman's correlation coefficient, shows high correlation values for all scoring functions, excluding the S_{IV} function. When it was used, the rankings calculated using the MARCOS method were significantly different.

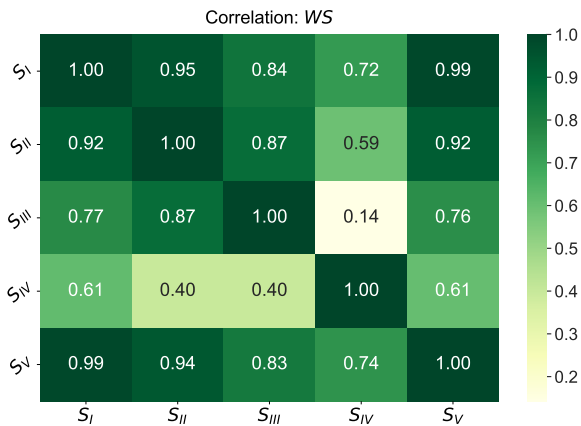


Fig. 6. WS correlation heatmap of MARCOS rankings for big decision matrix

In contrast, the similarity of the rankings calculated with WS coefficient, shown in Figure 6 also indicated that the S_{IV} function showed the least consistent results with the other techniques used. The strongest similarity of rankings could be observed for the pair of methods S_I and S_V , which is 0.99. In contrast, the lowest consistency of rankings is 0.14 for the pair of methods S_{III} and S_{IV} . It indicates a significant discrepancy, which confirms the importance of the influence of the used scoring function on the obtained results.

C. WS comparison

To generalize the results and examine the similarities between the scoring functions used, 1000 simulations were performed for randomly generated decision matrices. Each of the generated matrices was subjected to the techniques described earlier, and the resulting crisp matrices were used in a multi-criteria analysis using the MARCOS method. The figures and tables below show the values calculated for the similarities of the obtained rankings. The WS rank similarity coefficient determined their consistency.

Visualizations for selected scoring functions are presented below, together with tables describing selected statistics of the obtained data. Figure 7 shows the distribution of ranking similarity values for the simulations performed. The rankings obtained using the S_{II} function were compared with the other methods. It is worth noting that for the functions S_{III} and S_V , the similarity of the rankings was high and concentrated in a narrow area. It shows a high consistency in how IFS conversions to crisp values are performed, which translates into high reproducibility in evaluating alternatives.

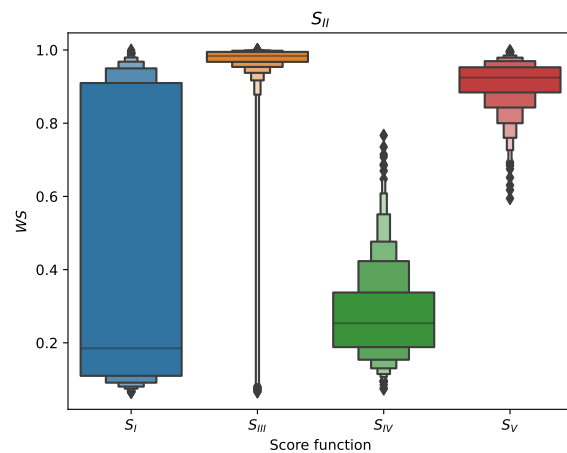


Fig. 7. Distribution of rankings similarity values using the S_{II} score function.

Table XV contains the statistics calculated from the simulations, including a comparison of the performance of the function S_I with the others. The variance and standard deviation were most negligible for the functions S_{III} and S_V , as confirmed by the data shown in Figure 7. On the other hand, the most significant standard deviation (0.384707) was seen when comparing the results obtained using the S_I function.

TABLE XV
STATISTICS FOR RESULTS OBTAINED USING THE S_{II} SCORE FUNCTION.

S_i	Standard deviation	Variance	Mean
S_I	0.384707	0.147999	0.446851
S_{III}	0.098412	0.009685	0.967222
S_{IV}	0.118155	0.013961	0.276578
S_V	0.060845	0.003702	0.910355

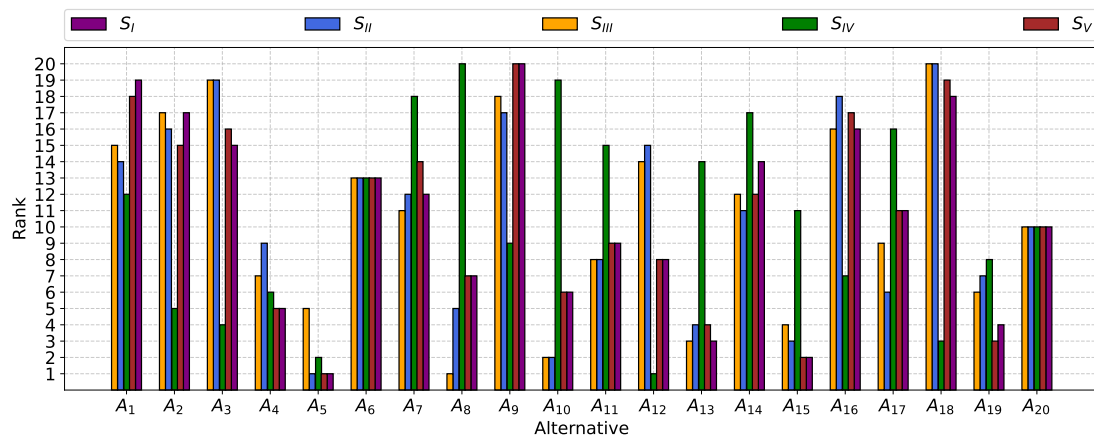


Fig. 4. MARCOS rankings for big decision matrix for S_I - S_V score functions.

Figure 8 shows the similarity distribution obtained for the comparison of results using the S_{III} function together with the other functions. As in the previous case, the highest similarity of rankings was observed for the functions S_{II} and S_V . In addition, the lowest consistency of results was noted when comparing with the ranking obtained using S_{IV} .

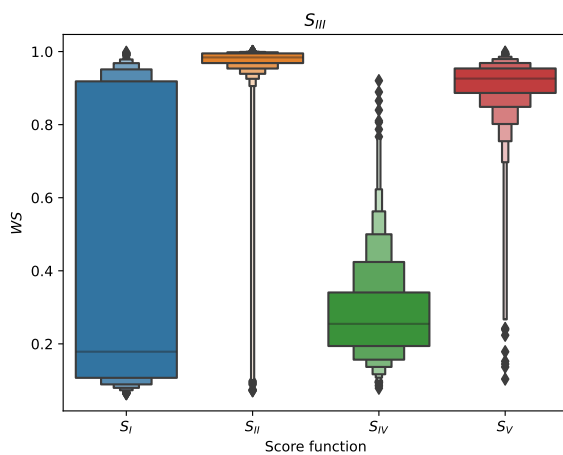


Fig. 8. Distribution of rankings similarity values using the S_{III} score function.

In turn, Table XVI describes the statistical values for the data obtained when comparing the rankings of the functions S_{III} with the others. The highest average ranking similarity value is 0.968083 for the method pair S_{III} and S_{II} . It demonstrates the high consistency of the results and shows that the two functions can be used interchangeably without much effect on the rankings in most cases.

TABLE XVI
STATISTICS FOR RESULTS OBTAINED USING THE S_{III} SCORE FUNCTION.

S_i	Standard deviation	Variance	Mean
S_I	0.391059	0.152927	0.452026
S_{II}	0.098297	0.009662	0.968083
S_{IV}	0.125697	0.015800	0.281541
S_V	0.093363	0.008717	0.906154

A visualization of the similarity distribution of the rankings obtained using the scoring function S_{IV} compared to the other functions is shown in Figure 9. It can be seen that none of the techniques used gives a strong rankings correlation. Instead, it causes the results obtained to vary, making it essential to bear in mind that the choice of scoring function directly impacts the results obtained.

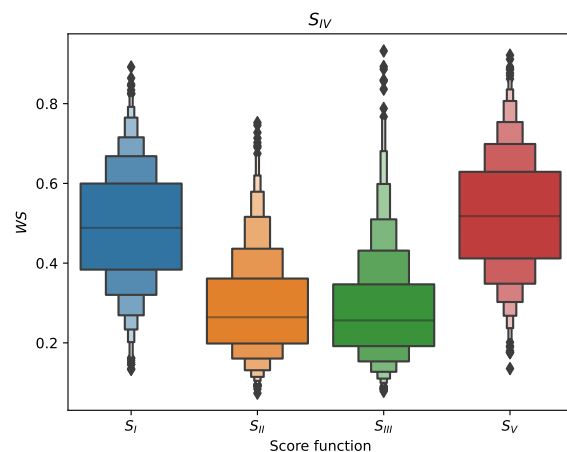


Fig. 9. Distribution of rankings similarity values using the S_{IV} score function.

The determined statistical features for the comparisons of the function S_{IV} with the others are listed in Table XVII. The average correlation value oscillates between a value of

0.284001 for feature S_{III} and 0.522578 for feature S_V . It shows that a low consistency of results is obtained regardless of the technique used. On the other hand, the standard deviation for the similarity of the rankings is similar across all functions. It shows that the quality of the correlation is also affected by the input data, which can improve or worsen the consistency of the rankings.

TABLE XVII
STATISTICS FOR RESULTS OBTAINED USING THE S_{IV} SCORE FUNCTION.

S_i	Standard deviation	Variance	Mean
S_I	0.145812	0.021261	0.491630
S_{II}	0.124837	0.015584	0.289953
S_{III}	0.133394	0.017794	0.284001
S_V	0.147851	0.021860	0.522578

The similarity results for the other functions used in the study, i.e., S_I and S_V , show that the first function gives a similar similarity of rankings to the other techniques. Still, it oscillates within a value of 0.4, indicating low consistency of the results. On the other hand, the second function shows a high similarity of performance together with the functions S_{II} and S_{III} . It confirms the trend of possible interchangeable use of these functions in converting IFS to crisp values in multi-criteria problems.

V. CONCLUSION

Decision-making appears in many parts of life, so developing this particular branch of technology is crucial. However, often in decision-making problems, the problem of uncertainty and fuzzy values arise, which makes standard methods inapplicable. For this reason, it is worth taking a closer look at the possibilities of defuzzification of such problems.

In the study carried out, five score functions that allow achieving crisp values from intuitionistic fuzzy sets were compared. Each of the functions allows obtaining completely different values, which ultimately will significantly influence the results of the rankings. The study showed that in the smaller problem, the functions S_I and S_{III} should be preferred in decision-making problems because of the high distinction of individual values between them. However, the more extensive problem and simulations for 1000 decision matrices showed that functions S_{II} , S_{III} and S_V proved to be the most coherent techniques. Moreover, those functions presented high similarity in resulting rankings rendering them equally capable.

In future studies, it would be meaningful to address this issue regarding the reference ranking to compare the performance of the used score functions to indicate their reliability in practical problems. In addition, it would verify the usefulness and effectiveness of presented score functions in the decision-making process, which is obligatory to obtain credible results. In addition, future research would need to consider real decision-making tasks.

ACKNOWLEDGMENT

The work was supported by the National Science Centre, Decision number 2021/41/B/HS4/01296 (B.K. and W.S).

REFERENCES

- [1] X. Gandibleux, "Multiple criteria optimization: state of the art annotated bibliographic surveys," 2006.
- [2] C. C. Aggarwal and S. Y. Philip, "A survey of uncertain data algorithms and applications," *IEEE Transactions on knowledge and data engineering*, vol. 21, no. 5, pp. 609–623, 2008.
- [3] P. Ziemia, J. Jankowski, and J. Wątróbski, "Online comparison system with certain and uncertain criteria based on multi-criteria decision analysis method," in *International Conference on Computational Collective Intelligence*. Springer, 2017, pp. 579–589.
- [4] D. Dubois and H. Prade, "Membership functions," in *Fuzzy Approaches for Soft Computing and Approximate Reasoning: Theories and Applications*. Springer, 2021, pp. 5–20.
- [5] V. Torra, "Hesitant fuzzy sets," *International journal of intelligent systems*, vol. 25, no. 6, pp. 529–539, 2010.
- [6] T. Senapati and R. R. Yager, "Fermatean fuzzy sets," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 2, pp. 663–674, 2020.
- [7] B. C. Cuong and V. Kreinovich, "Picture fuzzy sets," *Journal of Computer Science and Cybernetics*, vol. 30, no. 4, pp. 409–420, 2014.
- [8] P. Ejegwa, S. Akowe, P. Otene, and J. Ikyle, "An overview on intuitionistic fuzzy sets," *Int. J. Sci. Technol. Res.*, vol. 3, no. 3, pp. 142–145, 2014.
- [9] M. J. Khan, M. I. Ali, P. Kumam, W. Kumam, M. Aslam, and J. C. R. Alcantud, "Improved generalized dissimilarity measure-based vikor method for pythagorean fuzzy sets," *International Journal of Intelligent Systems*, vol. 37, no. 3, pp. 1807–1845, 2022.
- [10] P. Thakur, B. Kizielewicz, N. Gandotra, A. Shekhovtsov, N. Saini, A. B. Saeid, and W. Sařabun, "A new entropy measurement for the analysis of uncertain data in mcdm problems using intuitionistic fuzzy sets and copras method," *Axioms*, vol. 10, no. 4, p. 335, 2021.
- [11] S. Faizi, W. Sařabun, T. Rashid, S. Zafar, and J. Wątróbski, "Intuitionistic fuzzy sets in multi-criteria group decision making problems using the characteristic objects method," *Symmetry*, vol. 12, no. 9, p. 1382, 2020.
- [12] B. Gohain, R. Chutia, P. Dutta, and S. Gogoi, "Two new similarity measures for intuitionistic fuzzy sets and its various applications," *International Journal of Intelligent Systems*, 2022.
- [13] E. Szmidi, J. Kacprzyk, and P. Bujnowski, "Three term attribute description of atanassov's intuitionistic fuzzy sets as a basis of attribute selection," in *2021 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. IEEE, 2021, pp. 1–6.
- [14] N. X. Thao, "Some new entropies and divergence measures of intuitionistic fuzzy sets based on archimedean t-conorm and application in supplier selection," *Soft Computing*, vol. 25, no. 7, pp. 5791–5805, 2021.
- [15] S. Al-Humairi, A. Hizami, A. Zaidan, B. Zaidan, H. Alsattar, S. Qahtan, O. Albahri, M. Talal, A. Alamoody, and R. Mohammed, "Towards sustainable transportation: A pavement strategy selection based on the extension of dual-hesitant fuzzy multi-criteria decision-making methods," *IEEE Transactions on Fuzzy Systems*, 2022.
- [16] S.-M. Chen and J.-M. Tan, "Handling multicriteria fuzzy decision-making problems based on vague set theory," *Fuzzy sets and systems*, vol. 67, no. 2, pp. 163–172, 1994.
- [17] T.-Y. Chen, "A comparative analysis of score functions for multiple criteria decision making in intuitionistic fuzzy settings," *Information Sciences*, vol. 181, no. 17, pp. 3652–3676, 2011.
- [18] S. K. De, R. Biswas, and A. R. Roy, "An application of intuitionistic fuzzy sets in medical diagnosis," *Fuzzy sets and Systems*, vol. 117, no. 2, pp. 209–213, 2001.
- [19] A. Kharal, "Homeopathic drug selection using intuitionistic fuzzy sets," *Homeopathy*, vol. 98, no. 1, pp. 35–39, 2009.
- [20] Ž. Stević, D. Pamučar, A. Puška, and P. Chatterjee, "Sustainable supplier selection in healthcare industries using a new mcdm method: Measurement of alternatives and ranking according to compromise solution (marcos)," *Computers & Industrial Engineering*, vol. 140, p. 106231, 2020.
- [21] P. Ziemia, "Selection of electric vehicles for the needs of sustainable transport under conditions of uncertainty—a comparative study on fuzzy mcdm methods," *Energies*, vol. 14, no. 22, p. 7786, 2021.

Towards Sustainable Transport Assessment Considering Alternative Fuels Based on MCDA Methods

Jarosław Wątróbski, Aleksandra Bączkiewicz
Institute of Management, University of Szczecin
ul. Cukrowa 8, 71-004 Szczecin, Poland

Email: jaroslaw.watrobowski@usz.edu.pl, aleksandra.baczkiewicz@phd.usz.edu.pl

Abstract—Sustainable transport can contribute to many beneficial changes, such as reducing greenhouse gas emissions and pollutants into the atmosphere, improving the country’s energy security, and enhancing energy efficiency. Therefore, it is essential to provide a framework for reliable measurement of sustainable transport, enabling its evaluation in terms of diversity and the significance of renewable energy sources (RES). This paper presents a methodological framework for a multi-criteria assessment of sustainable transportation. The proposed framework is based on three multi-criteria decision analysis (MCDA) methods: SPOTIS (Stable Preference Ordering Towards Ideal Solution), ARAS (Additive Ratio Assessment), and TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution). The application of the proposed tool is demonstrated in an illustrative example of the assessment of European countries in terms of the share of alternative fuels in final energy consumption in road transport. The authors used the proposed framework to perform a comparative analysis considering three MCDA methods and two methods for determining the significance of evaluation criteria: equal and entropy weighting methods. The investigation has proven the practical suitability of the proposed tool in the problem of multi-criteria sustainable transport assessment. Furthermore, conducted analysis indicated that Sweden is characterized by the most sustainable transport in terms of significance and share of alternative fuels and RES and their diversification.

I. INTRODUCTION

CURRENT European environmental policy is focused on improving the ecological situation by reducing greenhouse gas (GHG) emissions [1]. The European Union’s goals also cover increasing the share of renewable energy sources (RES) in all fields, including road transport [2]. The road transport sector, dominated by petroleum-derived fuels, is the source of about 25% of total GHG emissions in Europe [3]. As a consequence, the use of alternative fuels that cause less atmosphere pollution and at the same time contribute to increased security of supply and optimal energy storage is widely promoted. The investigation presented in the paper [3] found that the increase in electric vehicles, gas-engine vehicles, and biofuel-powered vehicles contributes to more sustainable energy consumption and reduces carbon emissions.

The long-term goals of the European transport economy are to increase the use of alternative fuels. The strategy set by European Commission includes targets covering 60% reduction in carbon dioxide emissions from transport by 2050 compared to 1990 [4], [5], and 70% reduction of final oil consumption

by 2050. The mentioned strategy also covers reduced increase of congestion implied multimodal solutions and innovative technologies in transport improving energy efficiency [6]. This trend is promoted by technological development and targeted investments. This strategy is justified by the harmful effects of the combustion of conventional fuels on the environment, depletion of sources for conventional fuels, and the aim to reduce dependence on oil imports from countries beyond the European Union [7], [8]. Motor fuels consist of a group of liquid fuels, including gasoline, diesel, biofuels, and a group of gaseous fuels, such as liquefied petroleum gases (LPG). Gasoline is produced by the rectification of petroleum. The advantages of gasoline include its high calorific value, low sulfur content, and resistance to low temperatures. However, some of the disadvantages of gasoline include environmental pollution in its combustion process, depletion of petroleum reserves, and high-cost production. Increasing the popularity of alternative fuels regarding conventional fuels such as gasoline and diesel includes the promotion of fuels derived from sources such as natural gas, LPG, biofuels and hydrogen, and electricity [5]. Electric vehicles do not pollute the atmosphere with exhaust fumes. Moreover, when RES power them, they contribute to reducing carbon dioxide emissions and fossil fuel consumption. Electric vehicles are particularly advantageous in urban areas, where travel distances are usually short [2]. The development of electric vehicles contributes to sustainable transportation in urban areas due to limited emissions and noise. In addition, hybrid vehicles with an internal combustion engine and an electric motor are also being developed to help reduce toxins and carbon dioxide emissions in exhaust gases. Fuels that play a significant role in replacing petroleum-based fuels in road transportation are biogas and natural gas. Natural gas is advantageous compared to petroleum because of less environmental impact and lower cost. Natural gas has good potential as an alternative fuel in road transport due to its contribution to supply security and lower environmental impact than conventional fuels [9]. Currently, the market for natural gas vehicles, including compressed natural gas (CNG) and liquefied natural gas (LNG), is expected to grow, expanding the opportunities for various road participants and contributing to fuel market diversification. On the other hand, the long-term benefits of diesel oil, CNG, and LPG are limited

due to their fossil fuel character [2]. Bioethanol and biodiesel are other fuel types that contribute to making transport less dependent on oil. First generation bioethanol and biodiesel are obtained from agricultural crops, the second generation from lignocellulosic biomass and third generation biofuels are produced from algae. The popularity of biofuels is growing, influenced by environmental regulations [2]. In the case of biofuels, the constant development of technology makes it possible to produce them from different sources. In addition, biofuels are also blended with conventional fuels to reduce environmental pollution. Liquefied petroleum gas (LPG) is an alternative fuel derived from oil and natural gas but can also be derived from biomass. In terms of environmental impact, it is more eco-friendly than conventional fuels as it has lower emissions [10].

Indeed, as can be seen, reliable assessment requires the consideration of many different criteria [11], [12], which in the case of the problem analyzed in this paper are the different types of alternative fuels. Various alternative fuels contribute as a whole to the reduction of carbon dioxide and pollutants to the atmosphere and the reduction of dependence on imported petroleum-based fuels [6]. Moreover, the growing popularity of fuels from RES contributes to the realization of an essential principle of sustainable development, which is increasing the share of RES in all sectors of the economy [13], including road transport [14]. The need to simultaneously consider multiple fuels as criteria for the proposed framework for European country evaluation implies the application of multi-criteria decision analysis (MCDA) methods.

MCDA methods are widely used in transport assessment because they allow for building models with multiple criteria necessary to evaluate and consider them simultaneously. For example, the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) is widely popular among researchers and practitioners due to its wide range of applications in various fields and real-life decision problems [15]. The TOPSIS method was applied for transport assessment concerning energy and environmental efficiency divided into road, rail, and air transport sectors [6]. Multi-criteria evaluation of electric vehicles from the perspective of sustainable transportation priorities often considers comparative analysis involving several MCDA methods. In the framework proposed in [16], the authors used fuzzy extensions of three MCDA methods, including TOPSIS, Simple Additive Weighting (SAW), and Preference Ranking Organization Method for Enrichment of Evaluation (PROMETHEE), due to the necessity to regard the uncertainty occurring in the considered data on parameters and performance of evaluated vehicles, which is represented as interval or triangular fuzzy number (TFN). The vehicle rating for sustainable public transport presented in paper [17] employs a multi-criteria model built for evaluation using the ELECTRE (ELimination Et Choice Translating REality) TRI method. ELECTRE III was used to benchmark the buses recommended for the public transportation sector [18]. The authors of another work applied Multiattribute Utility Theory Approach (MAUT) method to determine consumer preferences

for alternative fuel vehicles [14]. In another research, the Analytical Hierarchy Process (AHP) was employed to evaluate alternative fuels in the context of sustainable transport [19]. Applying a multi-criteria decision analysis model based on the Additive Ratio Assessment (ARAS) method to assess alternative fuels for public transport was the main contribution of work [20]. The above literature review allows concluding that MCDA methods are widely and successfully used to evaluate sustainable transportation in many aspects. However, most research focuses on evaluating different vehicle technologies and fuels. On the other hand, attempts to rank countries concerning the share of alternative fuels in transport are rather limited. Therefore, it motivated the authors of this research to create a methodological framework based on MCDA methods that could support an information system for sustainable transport assessment, focusing on the share of alternative fuels in road transport.

This paper aims to present a multi-criteria approach, including the developed framework for measuring sustainable transport, particularly for assessing European countries considering sustainable energy consumption focused on alternative fuels in road transport. The objective of the research is to identify the country among the analyzed European countries where the share of alternative fuels is the most significant and diversified, which contributes to the reduction of greenhouse gas emissions and pollutants to the atmosphere and increases the country's independence from petroleum-derived fuels imports. The framework comprises a comparative analysis of results obtained using different MCDA methods to provide a reliable assessment [21], [22]. Another goal of the paper was to investigate the comparability of results obtained using three different MCDA methods based on distance measurement to reference solutions. Thus, the proposed approach involves employing three selected MCDA methods, including SPOTIS (Stable Preference Ordering Towards Ideal Solution), ARAS, and TOPSIS, for the multi-criteria evaluation of European countries regarding the share of alternative fuels in final Energy consumption in road transport. SPOTIS is a newly developed MCDA method that was introduced in 2020 [23]. The advantage of the SPOTIS method is resistance to the ranking reversal effect, which means that there is no ranking reversal when a particular alternative is removed or added from the evaluated set [24]. The mentioned advantage is possible because direct comparisons between alternatives are not required. In the SPOTIS approach, options are compared only with the ideal solution constructed by the decision-maker in the procedure of specification minimum and maximum bounds of each criterion which define multi-criteria problem to be solved. An additional advantage of the SPOTIS method is identifying the whole domain model due to building the ideal solution point defining the considered problem [23]. To confirm the results' reliability, the authors compared the results of the SPOTIS method with the results given by two other benchmark MCDA methods, ARAS and TOPSIS. The authors chose both benchmarking methods regarding a similar principle to the SPOTIS method, namely considering

the ideal solution in evaluating alternatives. The ARAS method evaluates alternatives by determining each option's degree of utility (efficiency) concerning the ideal alternative [25]. The TOPSIS method, on the other hand, takes into account their distance from the ideal and anti-ideal solutions in calculating the utility function values for each alternative [26]. In contrast to ARAS and TOPSIS, the SPOTIS method is more flexible because it allows the decision-maker to define the ideal solution independently instead of solely based on the data in the decision matrix [23]. Mentioned MCDA methods, besides providing a decision matrix containing performance values against the criteria, also require assigning a value of each criterion importance to the decision-maker, i.e., a weight. A strategic approach to fulfilling the long-term needs of all modes of transportation is recommended to be based on a full suite of alternative fuels without preference for particular types [9], [27]. The authors assigned equal weights to the criteria with this fact in mind. However, the authors also included in performed analysis objective weights determined by the Entropy method for a more comprehensive and reliable research procedure.

The rest of the paper is organized as follows. Section II provides basic assumptions and mathematical formulas of methods applied in this research. Next, in section III the practical problem of European countries' assessment regarding the share of alternative fuels in consumption in final energy consumption in road transport. Then, in section IV research results are presented. In section V discussion of obtained results is provided. Finally, in the last section VI the summary and conclusions are given, and future work directions are drawn.

II. METHODOLOGY

This section provides the basics and main assumptions of the particular MCDA methods employed in this research and other supporting techniques as criteria weighting methods and correlation coefficients for determining obtained rankings consistency for benchmarking analysis.

A. The SPOTIS Method

The subsequent stages of the SPOTIS (Stable Preference Ordering Towards Ideal Solution) method are given based on [23].

Step 1. Define the MCDA problem by determining the bounds containing the minimum and maximum performance values included in evaluated decision matrix $S = [s_{ij}]_{m \times n}$ for each criterion. The minimum and maximum bounds For each criterion $C_j (j = 1, 2, \dots, n)$ is determined respectively by S_j^{min} and S_j^{max} .

Step 2. Determine the Ideal Solution Point (ISP) represented by S^* . When for the criterion C_j larger score value is preferable, then the ISP for criterion C_j is $S_j^* = S_j^{max}$. From the other side when for the criterion C_j lower score value is favored, then the ISP for criterion C_j is $S_j^* = S_j^{min}$. The ideal multi-criteria best solution S^* is denoted by coordinates $(S_1^*, S_2^*, \dots, S_n^*)$.

Step 3. Determine the normalized distance values d_{ij} based on ISP for each considered alternative A_i according to Equation (1).

$$d_{ij}(A_i, s_j^*) = \frac{|S_{ij} - S_j^*|}{|S_j^{max} - S_j^{min}|} \quad (1)$$

Step 4. Calculate of the weighted normalized averaged distance values for each alternative A_i as Equation (2) shows

$$d(A_i, s^*) = \sum_{j=1}^n w_j d_{ij}(A_i, s_j^*) \quad (2)$$

where w_j represents the weight of j th criterion.

Step 5. Create ranking of evaluated alternatives by sorting $d(A_i, s^*)$ values in ascending order. Alternative with the lowest $d(A_i, s^*)$ value is the best scored option.

B. The ARAS Method

The following stages of the ARAS method are presented below, based on [25].

Step 1. Normalize the decision matrix using the Sum normalization method applying Equation (3) for benefit criteria and Equation (4) for cost criteria.

$$r_{ij} = \frac{x_{ij}}{\sum_{i=1}^m x_{ij}} \quad (3)$$

$$r_{ij} = \frac{\frac{1}{x_{ij}}}{\sum_{i=1}^m \frac{1}{x_{ij}}} \quad (4)$$

where $X = [x_{ij}]_{m \times n}$ represents the decision matrix containing performance values of m alternatives in respect to n evaluation criteria.

Step 2. Calculate the weighted normalized decision matrix $D = [d_{ij}]_{m \times n}$ according to Equation (5)

$$d_{ij} = r_{ij} w_j \quad (5)$$

where w_j denotes j th criteria weight values.

Step 3. Compute the optimality function S_i for each i th alternative as Equation 6 presents.

$$S_i = \sum_{j=1}^n d_{ij} \quad (6)$$

Step 4. Calculate the utility value U_i for each i th alternative according to Equation (7)

$$U_i = S_i / S_o \quad (7)$$

where S_o denotes the optimality function value for the optimal alternative. U_i values are in the range from 0 to 1. The option which has the highest U_i value is regarded as the best scored alternative. Thus, the ranking of the ARAS method is constructed in descending order according to U_i values.

C. The TOPSIS Method

The successive steps of the TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution) method are demonstrated as follows, based on [26].

Step 1. Normalize the decision matrix with performance values by chosen normalization technique, for example, the Minimum-Maximum normalization method, as performed in this research. Another normalization method can also be employed for this aim. In the Minimum-Maximum normalization r_{ij} normalized values of decision matrix are calculated by Equation (8) for benefit criteria and (9) for cost criteria.

$$r_{ij} = \frac{x_{ij} - \min_j(x_{ij})}{\max_j(x_{ij}) - \min_j(x_{ij})} \quad (8)$$

$$r_{ij} = \frac{\max_j(x_{ij}) - x_{ij}}{\max_j(x_{ij}) - \min_j(x_{ij})} \quad (9)$$

where $X = [x_{ij}]_{m \times n}$ represents the decision matrix containing performance values of m alternatives in respect to n evaluation criteria.

Step 2. Calculate the weighted normalized decision matrix as Equation (10) presents.

$$v_{ij} = w_j r_{ij} \quad (10)$$

where w_j denotes j th criteria weight values.

Step 3. Determine the Positive Ideal Solution (PIS) using Equation (11) and Negative Ideal Solution (NIS) by Equation (12). PIS includes the maximum values of the weighted normalized decision matrix, while in NIS, its minimal values are contained. Since the normalization of the decision matrix was applied in the previous step, there is no necessity to separate the criteria into profit and cost types.

$$v_j^+ = \{v_1^+, v_2^+, \dots, v_n^+\} = \{\max_j(v_{ij})\} \quad (11)$$

$$v_j^- = \{v_1^-, v_2^-, \dots, v_n^-\} = \{\min_j(v_{ij})\} \quad (12)$$

Step 4. Calculate the distance from PIS (13) and NIS (14) of each alternative. The default measure for distance determination in TOPSIS method is Euclidean distance.

$$D_i^+ = \sqrt{\sum_{j=1}^n (v_{ij} - v_j^+)^2} \quad (13)$$

$$D_i^- = \sqrt{\sum_{j=1}^n (v_{ij} - v_j^-)^2} \quad (14)$$

Step 5. Calculate the utility function value for each evaluated alternative as Equation (15) shows. The C_i value is within the range from 0 to 1, and the alternative with the highest C_i value is the most preferred. Thus, the ranking of the TOPSIS method is constructed by descending ordering of alternatives according to their utility function values.

$$C_i = \frac{D_i^-}{D_i^- + D_i^+} \quad (15)$$

D. The Equal Weighting Method

The equal weighting method is the simplest technique for determining criteria significance values. However, for several MCDA problems assigning equal weights to evaluation criteria is appropriate. Equal weights are determined as Equation (16) shows.

$$w_j = 1/n \quad (16)$$

where n represents number of evaluation criteria.

E. The Entropy Weighting Method

The entropy weighting method is an objective weighting technique based on Shannon's entropy theory. Shannon entropy performs an important role in information theory. For example, entropy is used to measure the information included in data, which is contained in a two-dimensional decision matrix in the case of MCDA problems [1]. The following stages of the Entropy weighting method are given as follows, based on [1].

Step 1. Normalize the decision matrix using sum normalization method to obtain normalized decision matrix $P = [p_{ij}]_{m \times n}$ where $i = 1, 2, \dots, m$ and $j = 1, 2, \dots, n$, m represents number of alternatives and n denotes number of evaluation criteria.

Step 2. Calculate the entropy value E_j for each j th criterion as Equation (17) shows.

$$E_j = -\frac{\sum_{i=1}^m p_{ij} \ln p_{ij}}{\ln m} \quad (17)$$

Step 3. Calculate d_j value according to Equation (18).

$$d_j = 1 - E_j \quad (18)$$

Step 4. Calculate the entropy weights for each j th criterion as Equation (19) shows.

$$w_j = \frac{d_j}{\sum_{j=1}^n d_j} \quad (19)$$

F. The Weighted Spearman Rank Correlation Coefficient

The r_w correlation coefficient is determined to compare two rankings x and y according to Equation (20). N denotes a number of rank values x_i and y_i [1].

$$r_w = 1 - \frac{6 \sum_{i=1}^N (x_i - y_i)^2 ((N - x_i + 1) + (N - y_i + 1))}{N^4 + N^3 - N^2 - N} \quad (20)$$

G. The Spearman Rank Correlation Coefficient

The Spearman Rank Correlation Coefficient is computed to determine the correlation between two rankings x and y according to Equation (21)

$$r_s = 1 - \frac{6 \cdot \sum_{i=1}^N (x_i - y_i)^2}{N \cdot (N^2 - 1)} \quad (21)$$

where N denotes size of vector x and y [28].

III. THE PRACTICAL PROBLEM OF EUROPEAN COUNTRIES' ASSESSMENT REGARDING ALTERNATIVE FUELS IN TOTAL FINAL ENERGY CONSUMPTION IN ROAD TRANSPORT

This paper aims to assess 32 selected European countries regarding sustainable energy consumption considering alternative fuels in road transport. For this purpose, the authors developed the framework based on annual data on final energy consumption in road transport provided by Eurostat, considering different alternative fuel types [29] (accessed on 2 May 2022). Furthermore, to analyze the up-to-date situation, the authors gathered the most recent data available in the Eurostat database for 2020. This Eurostat frame excludes off-road use of fuels from road transport (for example, cranes and excavators at construction sites, harvesters, and tractors at fields). However, it is included in the respective consumption sector. Road transport includes passenger and freight transport, domestic and international transport, urban and intercity transport performed on public road networks, and publicly accessible private road network, including both the free and the paid part of the road network systems. The authors assessed sustainable energy consumption in road transport by incorporating the percentage share of each considered alternative fuel type in the annual total final energy consumption in road transport. Table I presents alternative fuel types playing the role of evaluation criteria in the proposed framework. Each criterion has to be maximized because the objective is to increase the share of alternative fuels in total road transport fuel consumption. Due to the recommended lack of preference for particular alternative fuel types, each criterion has the same significance value represented by equal weights. The last two columns of Table I provide the significance values of the criteria of the proposed evaluation framework, i.e., the different types of alternative fuels considered.

TABLE I
EVALUATION CRITERIA INCLUDING SHARE OF ALTERNATIVE FUEL TYPES IN TOTAL FINAL ENERGY CONSUMPTION IN ROAD TRANSPORT.

C_j	Criterion name	Unit	Aim	Equal Weight	Entropy Weight
C_1	Blended biodiesels	[%]	Max	0.1111	0.0103
C_2	Liquefied petroleum gases	[%]	Max	0.1111	0.0754
C_3	Blended biogasoline	[%]	Max	0.1111	0.0198
C_4	Natural gas	[%]	Max	0.1111	0.0543
C_5	Pure biodiesels	[%]	Max	0.1111	0.1875
C_6	Biogases	[%]	Max	0.1111	0.1370
C_7	Electricity	[%]	Max	0.1111	0.0770
C_8	Pure biogasoline	[%]	Max	0.1111	0.2403
C_9	Other liquid biofuels	[%]	Max	0.1111	0.1985

Data for the mentioned criteria were collected from the Eurostat database available at the link [29] for 32 selected European countries listed in Table II. The value of each criterion is provided in the Eurostat database in the unit called a Thousand tonnes of oil equivalent (TOE). For a representative and reliable countries assessment in terms of sustainable fuel consumption in transport sustainability, the authors calculated the share of each fuel as a percentage of

total annual fuel consumption based on the available values. Such an approach allows for an individual approach for each country that adequately considers the needs and capacities of countries resulting from independent aspects such as geography, area, and population size.

Figure 1 displays, in the form of a stacked column chart, the percentage of alternative fuels in final Energy consumption in road transport in 2020 for which investigation was performed. The chart analysis allows us to observe the largest share of alternative fuels as a whole in the considered domain for Sweden (SE). Blended biodiesels represent the most significant part of this share (C_1). Apart from that, pure biodiesels (C_5), blended biogasoline (C_3), biogases (C_6), and electricity (C_7) account for a significant share. The chart provided demonstrates the dominance of blended biodiesels (C_1) and LPG (C_2) share among the countries assessed. The countries where energy consumption from electricity in road transport is most noticeable are Norway (NO), Iceland (IS), and Sweden (SE). However, an evaluation based only on the cumulative values of individual fuels is insufficient because it does not allow for simultaneous consideration of the degree of diversification of fuels and decision-makers preferences concerning particular fuels. Therefore, to consider the mentioned aspects, a framework employing MCDA methods is recommended [30].

IV. RESULTS

This section presents the results of each MCDA method individually for the equal and entropy evaluation criteria weights. The results include MCDA utility function values for each alternative, rankings constructed based on them, and analysis of obtained rankings convergence represented by correlation coefficient values.

A. Results for the Equal Weighting Method

Table III contains utility function values and rankings received for evaluated countries concerning equal weights assigned to considered criteria with SPOTIS, ARAS, and TOPSIS methods. For the SPOTIS method, the alternative that received the lowest utility function value is the best-ranked alternative. On the other hand, for the ARAS and TOPSIS methods, the alternative that has the highest utility function value is the best option.

It can be observed that when the criteria are assigned equal weights, all MCDA methods used in this research indicated Sweden (SE) as the country with the most significant share of alternative fuels in final Energy consumption in road transport. Another well-scored country is Norway (NO), ranked second in all rankings. Norway was ranked better than Bulgaria (BG) and Ukraine (UA), although its overall share of alternative fuels is lower. Norway was nevertheless ranked second because it has a more diversified share of alternative fuels than Bulgaria and Ukraine. Therefore, diversification of alternative fuels in final energy consumption in transport is promoted, and MCDA methods enable appropriate reflection of this fact. Norway's share of alternative fuels in road transport consists of a mix covering six different alternative fuel types, namely

TABLE II
DECISION MATRIX WITH PERCENTAGE SHARES OF ALTERNATIVE FUELS IN FINAL ENERGY CONSUMPTION IN ROAD TRANSPORT IN 2020.

Country	Code	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9
Austria	AT	4.5648	0.0426	0.7669	0.2713	0.3467	0.0047	0.0162	0	0
Belgium	BE	7.7250	0.6543	1.7500	0.5747	0	0	0.1553	0	0
Bulgaria	BG	4.6534	13.2909	0.8475	2.7555	0	0	0.0998	0	0
Croatia	HR	3.4274	3.1351	0.0402	0.1629	0.0085	0	0.0128	0	0
Cyprus	CY	4.0226	0.0733	0.1059	0	0	0	0	0	0
Czechia	CZ	4.9672	1.2723	1.0810	1.2599	0	0.0193	0.1005	0	0
Denmark	DK	4.6560	0	2.1625	0.2147	0	0	0.2227	0	0
Estonia	EE	4.2500	1.3120	0.8033	2.0164	0	1.0882	0.1604	0	0
Finland	FI	8.2455	0	2.4812	0.3792	0	0.1117	0.2963	0	0
France	FR	5.7580	0.1086	1.5421	0.5374	0	0	0.1014	0	0
Germany	DE	5.2409	0.5146	1.4152	0.2731	0.0093	0.1558	0.0784	0	0.0014
Greece	EL	3.0211	4.2903	1.3879	0.4014	0	0	0.0371	0	0
Hungary	HU	4.5742	0.3356	1.9646	0.1818	0	0	0.1229	0	0
Iceland	IS	4.6093	0	2.3228	0	0	0.5875	0.8353	0	0
Ireland	IE	4.6710	0.0253	0.5853	0	0	0	0.1042	0	0
Italy	IT	4.6143	5.3319	0.0728	2.8625	0	0.0001	0.0607	0	0
Latvia	LV	3.1032	4.3335	1.2623	0.0529	0	0	0.2314	0	0
Lithuania	LT	4.1574	4.7534	0.7719	0.4112	0	0	0.1708	0	0
Luxembourg	LU	7.7657	0.0116	0.8351	0	0	0	0.0896	0	0.0013
Malta	MT	7.0464	0.3755	0	0	0	0	0.0969	0	0
Netherlands	NL	3.3488	1.2979	2.5357	0.6489	0	0	0.6481	0	0
Norway	NO	10.5502	0.1851	1.1836	0.7354	0	0.6206	2.9069	0	0
Poland	PL	3.9713	9.0431	0.8682	0.0925	0.0920	0	0.0304	0	0
Portugal	PT	4.8778	0.6410	0.1336	0.3187	0.0294	0	0.0186	0	0
Romania	RO	5.4875	1.4321	1.4831	0	0.8511	0	0.0576	0	0
Serbia	RS	0	3.5315	0	0.9394	0	0	0	0	0
Slovakia	SK	5.5505	0	1.1137	0	0	0	0.0886	0	0
Slovenia	SI	5.4665	0.5868	0.5157	0.2236	0	0	0.0118	0	0
Spain	ES	5.3262	0.3110	0.3583	1.1540	0.1457	0	0.0725	0.0012	0
Sweden	SE	14.8589	0	1.5113	0.1382	4.6082	1.4162	0.6828	0.0509	0
Ukraine	UA	0	19.8763	0.7555	0.2694	0	0	0	0	0
United Kingdom	UK	3.2833	0.1745	1.0015	0	0	0	0.0817	0	0

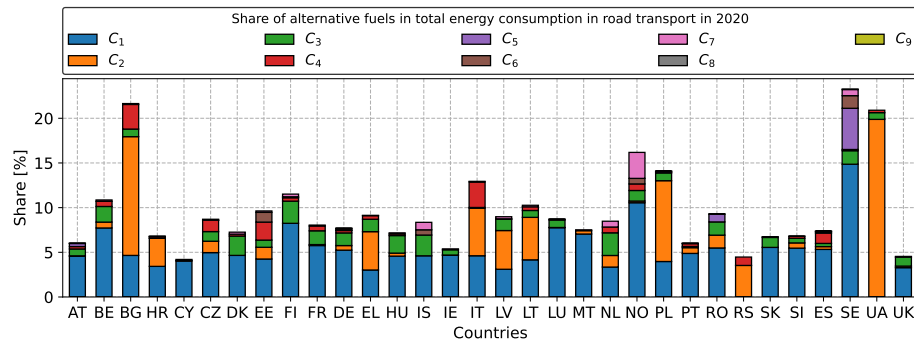


Fig. 1. Shares of alternative fuels in annual final energy consumption of evaluated countries in road transport in 2020.

C_1 , C_2 , C_3 , C_4 , C_6 , and C_7 . A particularly favorable result is implicated by the highest percentage of energy consumption of C_7 by Norway. For Bulgaria (BG), the share of four alternative fuel types (C_1 , C_2 , C_3 , and C_4) and the trace share of C_7 are noticeable. In the case of Ukraine (UA), the contribution of only three types of alternative fuels covering C_2 , C_3 , and C_4 is evident. Bulgaria (BG) was ranked third in SPOTIS and TOPSIS, while it was ranked sixth in the ARAS ranking. Bulgaria (BG) was thus rated better than Ukraine (UA), supported by a more diversified alternative fuel mix. The worst-rated country by all MCDA methods applied in the

presented research was Cyprus (CY), which has a low share of only three alternative fuels comprising C_1 , C_2 , and C_3 , with the most significant share of C_1 . Besides the two top places and the last place in the obtained rankings of the evaluated countries, some divergences are noticeable depending on the MCDA method used. The divergences occurring in each ranking are visualized in the column chart in Figure 2. Due to the observed differences in rankings, objective measures of convergence of compared rankings were applied to establish the degree of divergence of particular rankings, which are two ranking correlation coefficients: the Weighted Spearman Rank

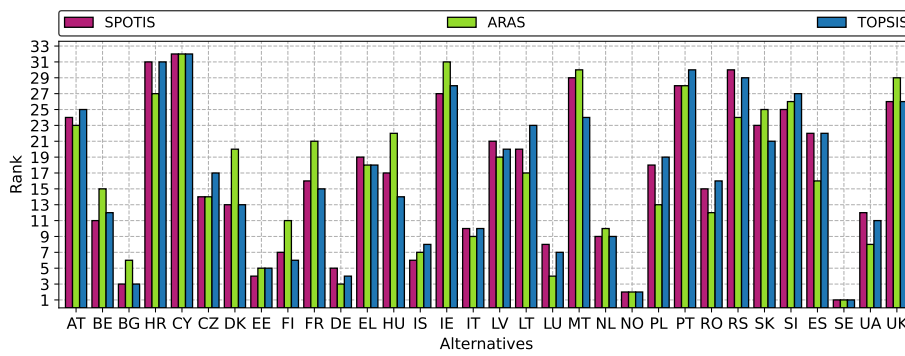


Fig. 2. Comparison of rankings obtained using three different MCDA methods for Equal criteria weights.

Correlation Coefficient r_w and the Spearman Rank Correlation Coefficient r_s .

TABLE III
RESULTS OF MCDA METHODS FOR EQUAL CRITERIA WEIGHTS.

Country	Utility function value			Rank		
	SPOTIS	ARAS	TOPSIS	SPOTIS	ARAS	TOPSIS
AT	0.9121	0.0415	0.1396	24	23	25
BE	0.8336	0.0617	0.2544	11	15	12
BG	0.7430	0.1457	0.3402	3	6	3
HR	0.9481	0.0273	0.0910	31	27	31
CY	0.9649	0.0114	0.0862	32	32	32
CZ	0.8541	0.0641	0.2113	14	14	17
DK	0.8536	0.0511	0.2542	13	20	13
EE	0.7559	0.1638	0.3178	4	5	5
FI	0.7948	0.0793	0.3073	7	11	6
FR	0.8640	0.0489	0.2186	16	21	15
DE	0.7588	0.2011	0.3281	5	3	4
EL	0.8756	0.0535	0.1925	19	18	18
HU	0.8661	0.0452	0.2367	17	22	14
IS	0.7857	0.1205	0.3024	6	7	8
IE	0.9353	0.0228	0.1217	27	31	28
IT	0.8191	0.1014	0.2915	10	9	10
LV	0.8864	0.0528	0.1787	21	19	20
LT	0.8860	0.0567	0.1577	20	17	23
LU	0.7963	0.1723	0.3059	8	4	7
MT	0.9415	0.0228	0.1427	29	30	24
NL	0.8066	0.0854	0.2939	9	10	9
NO	0.6799	0.2265	0.3812	2	2	2
PL	0.8747	0.0676	0.1908	18	13	19
PT	0.9403	0.0253	0.1104	28	28	30
RO	0.8632	0.0744	0.2132	15	12	16
RS	0.9438	0.0365	0.1157	30	24	29
SK	0.9063	0.0305	0.1727	23	25	21
SI	0.9241	0.0275	0.1329	25	26	27
ES	0.8891	0.0579	0.1715	22	16	22
SE	0.4579	0.5999	0.5236	1	1	1
UA	0.8453	0.1040	0.2792	12	8	11
UK	0.9275	0.0241	0.1389	26	29	26

High values of these coefficients close to 1 indicate high convergence of the compared rankings. Figure 3 displays the r_w and r_s coefficient values calculated for the pairwise comparisons of the obtained rankings. The correlation coefficient r_w calculated for the compared SPOTIS and TOPSIS rankings has the highest value, equal to 0.9887. Furthermore, the correlation coefficient r_w between SPOTIS and ARAS

rankings is also high, 0.9417. The lowest correlation of 0.9254 was observed for comparing ARAS and TOPSIS rankings. The values of the second correlation coefficient r_s are similar. High correlation values for comparisons of SPOTIS ranking with rankings provided by benchmarking methods TOPSIS and ARAS confirm the results' reliability.

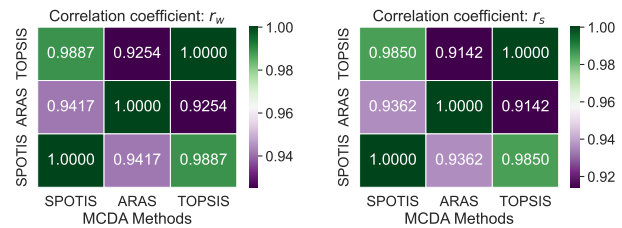


Fig. 3. Correlation between rankings for Equal criteria weights.

B. Results for the Entropy Weighting Method

This section presents the second part of the research for the evaluation criteria weights determined by the objective Entropy method. This part of the analysis was conducted to objectify the research results. Objective criterion weighting techniques are used to determine criterion weights based on the outcomes of mathematical models. Objective weighting techniques are useful when determining reliable subjective weights by the decision-maker is not possible, for example, due to the lack of experts with the necessary knowledge of the multi-criteria problem to be solved. Figure 4 displays a chart comparing SPOTIS, ARAS, and TOPSIS rankings for entropy weights visually. Table IV contains the utility function values and rankings of applied MCDA methods obtained for evaluated countries. The leader of all rankings is Sweden (SE), as it was noticed for the use of equal criteria weights. Germany (DE) took second place in all rankings employing Entropy criteria weights, which is different from equal criteria weights. Germany was ranked fifth for equal weights by the SPOTIS method, fourth by the TOPSIS method, and third by the ARAS method.

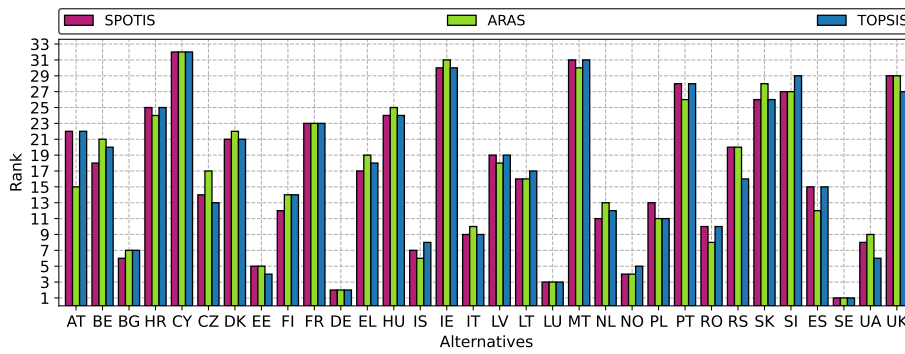


Fig. 4. Comparison of rankings obtained using three different MCDA methods for Entropy criteria weights.

TABLE IV
RESULTS OF MCDA METHODS FOR ENTROPY CRITERIA WEIGHTS.

Country	Utility function value			Rank		
	SPOTIS	ARAS	TOPSIS	SPOTIS	ARAS	TOPSIS
AT	0.9706	0.0222	0.0396	22	15	22
BE	0.9635	0.0133	0.0447	18	21	20
BG	0.8848	0.0555	0.1555	6	7	7
HR	0.9816	0.0096	0.0300	25	24	25
CY	0.9961	0.0010	0.0071	32	32	32
CZ	0.9548	0.0188	0.0612	14	17	13
DK	0.9699	0.0105	0.0439	21	22	21
EE	0.8380	0.1028	0.2285	5	5	4
FI	0.9491	0.0225	0.0592	12	14	14
FR	0.9707	0.0100	0.0391	23	23	23
DE	0.7622	0.2123	0.3643	2	2	2
EL	0.9622	0.0162	0.0498	17	19	18
HU	0.9735	0.0086	0.0389	24	25	24
IS	0.8997	0.0644	0.1411	7	6	8
IE	0.9893	0.0039	0.0151	30	31	30
IT	0.9201	0.0366	0.1267	9	10	9
LV	0.9644	0.0174	0.0479	19	18	19
LT	0.9607	0.0193	0.0501	16	16	17
LU	0.7971	0.1874	0.3471	3	3	3
MT	0.9911	0.0040	0.0138	31	30	31
NL	0.9435	0.0261	0.0683	11	13	12
NO	0.8318	0.1157	0.2065	4	4	5
PL	0.9499	0.0281	0.0811	13	11	11
PT	0.9854	0.0068	0.0182	28	26	28
RO	0.9430	0.0503	0.0867	10	8	10
RS	0.9688	0.0153	0.0524	20	20	16
SK	0.9851	0.0045	0.0235	26	28	26
SI	0.9854	0.0050	0.0177	27	27	29
ES	0.9570	0.0280	0.0567	15	12	15
SE	0.3925	0.6935	0.5966	1	1	1
UA	0.9136	0.0476	0.1595	8	9	6
UK	0.9871	0.0042	0.0204	29	29	27

Third place in all rankings was achieved by Luxembourg (LU), which in the case of applying equal weights ranked fourth in ARAS, seventh in TOPSIS, and eighth in SPOTIS. The better performance of Germany and Luxembourg is reflected in the fact that they are the only of the evaluated countries that have a C_9 share in final energy consumption in road transport. In the case of entropy weights, this is the second most important evaluation criterion. The results show that the determined criterion weights are critical for the MCDA evaluation results. For the weights determined by the

entropy method in the problem analyzed in this paper, the highest weight was assigned to criterion C_8 . It is reflected in Spain's better performance (ES) for entropy weights than equal weights. Spain is the only country besides Sweden with a C_8 share in final energy consumption in road transport. Cyprus (CY) was ranked last for entropy criteria weights, as was the case with assigning equal importance to the evaluation criteria.

It can be observed that the convergence of obtained rankings is higher for entropy weights than for equal weights. The high convergence of the rankings is confirmed in Figure 5, displaying the values of correlation coefficients r_w and r_s .

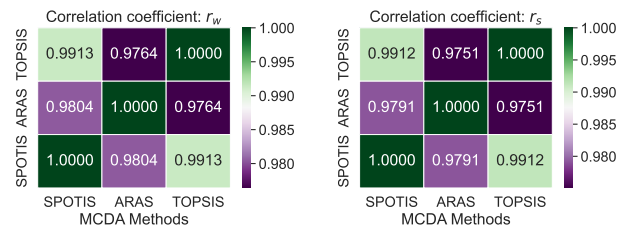


Fig. 5. Correlation between rankings for Entropy criteria weights.

The calculated correlation coefficients are the highest for comparing SPOTIS and TOPSIS rankings (r_w equal to 0.9913 and r_s equal to 0.9912). However, there was also a strong correlation between the SPOTIS and ARAS rankings (r_w equal to 0.9804 and r_s equal to 0.9791). The lowest correlation was observed between the ARAS and TOPSIS rankings. Conducted research showed that the strength of correlation between the determined country rankings for entropy weights is analogous to equal weights. The highest similarity is noticed in the case of SPOTIS and TOPSIS rankings.

The performed investigation proved that Sweden has the most significant and diversified share of alternative fuels in final energy consumption in road transport among 32 European countries evaluated in this research. Furthermore, the high score obtained by this country was confirmed by all MCDA methods, applying both criteria weighting methods.

V. DISCUSSION

It is difficult to compare various alternative fuels because different conflicting goals characterize them. For example, the popularization of electric vehicles contributes to reducing greenhouse gas emissions and may also increase water consumption. Therefore, the sustainability framework recommends evaluating technologies or alternatives by considering multiple dimensions to promote sustainability in each dimension [31], [32], [33]. Many countries intend to replace fossil fuel vehicles with electric vehicles soon. For example, among these countries is Norway, which ranked a high second in this study for equal criteria weights in SPOTIS, ARAS, and TOPSIS and has the largest share of electricity in final energy consumption in road transport [34]. Although the transition to electric vehicles seems promising for sustainable transportation, there are some difficulties such as high purchasing costs, exploitation problems, limited range and long charging times for vehicles, and limited availability of necessary charging infrastructure. Due to the mentioned aspects, it is essential to consider different types of alternative fuels in the sustainable development of transportation. Sweden is a representative example of a country characterized by a high diversification of types of alternative fuels in road transport, identified as the leader of the rankings in this research.

In Sweden, an important goal is the decarbonization process involving increasing the share of alternative fuels in road transport [35]. Therefore, there is considerable interest in exploiting alternative fuels in road transport in Sweden. Biogas is recognized as an alternative fuel in Sweden with significant environmental and social advantages. The technological maturity of biogas is noticeable in the area of biomethane buses. Positive aspects of biogas that public organizations appreciate in Sweden are energy security, nutrient recovery, and reduced environmental pollution. In addition, the Swedish regional transport government contributes to popularizing renewable fuels in the bus fleet [36]. Many public organizations in Sweden focused on bus transportation tendering processes want to contribute to sustainability improvements, including a transformation to reduce fossil fuels and increase renewables in bus transportation. Buses are dominant in public transport in Sweden. Bus fuels in public transport have seen a noticeable transformation over the past two decades. At the beginning of the 21st century, the bus fleet used fossil fuels almost entirely, while in 2017, more than 60% of buses used fuels produced from RES. In Sweden, the transformation towards more alternative fuels for road transport is mainly taking place on a regional level and includes bioethanol, biomethane, biodiesel, HVO (Hydrotreated Vegetable Oil), and most recently, electric buses. Electric buses are very popular, and their role is expected to increase in the future, especially in city centers [37]. The results confirm that diversification and development of alternative fuels at multiple levels contribute to a good evaluation of a country using an evaluation framework that includes different MCDA methods and criteria weighting techniques, as illustrated by an example of Sweden.

VI. CONCLUSIONS

The aim of this paper was to present a methodological framework that can be useful in supporting an information system for measurement and assessment of sustainable transport focused on the share of alternative fuels in final energy consumption in road transport. The application of the proposed framework was demonstrated in the illustrative example of the assessment of 32 selected European countries regarding the importance of the share and diversification of alternative fuels in road transport. The research results proved the usefulness of the presented approach in the analyzed problem of sustainable transport assessment. The applied approach indicated Sweden as the best-evaluated country concerning the criteria in the demonstrated evaluation framework. The obtained results showed that the MCDA-based approach has an advantage over simple aggregation methods. It allows a multidimensional assessment with simultaneous consideration of multiple criteria. Such an approach is compatible with the principle of diversification of alternative fuels in sustainable transport. Moreover, models based on MCDA methods enable prioritization of individual fuel types by assigning them significance values that may be equal or determined by objective or subjective weighting methods.

The results encourage the follow-up of research work in the scope of multi-criteria evaluation of sustainable transport considering different fuel types. Further work includes research on the influence of other methods of prioritizing assessment criteria on the results and exploring other MCDA methods, such as PROMETHEE II, which provides different preference functions [38]. Another interesting research direction is the temporal assessment of sustainable transport, considering the dynamics of performance changes in the analyzed time interval. Further work directions also include consideration of the economic aspects of alternative fuels and the level of self-sufficiency in the context of alternative fuel supply.

ACKNOWLEDGMENT

The work was supported by the project financed within the framework of the program of the Minister of Science and Higher Education under the name "Regional Excellence Initiative" in the years 2019-2022, Project Number 001/RID/2018/19; the amount of financing: PLN 10.684.000,00 (J.W. and A.B.).

REFERENCES

- [1] A. Bączkiewicz, B. Kizielewicz, A. Shekhovtsov, M. Yelmikheiev, V. Kozlov, and W. Sałabun, "Comparative analysis of solar panels with determination of local significance levels of criteria using the MCDM methods resistant to the rank reversal phenomenon," *Energies*, vol. 14, no. 18, p. 5727, 2021. doi: <https://doi.org/10.3390/en14185727>
- [2] K. G. Tsita and P. A. Pilavachi, "Decarbonizing the Greek road transport sector using alternative technologies and fuels," *Thermal Science and Engineering Progress*, vol. 1, pp. 15–24, 2017. doi: <https://doi.org/10.1016/j.tsep.2017.02.003>
- [3] J. L. Osorio-Tejada, E. Llera-Sastresa, and S. Scarpellini, "Liquefied natural gas: Could it be a reliable option for road freight transport in the EU?" *Renewable and Sustainable Energy Reviews*, vol. 71, pp. 785–795, 2017. doi: <https://doi.org/10.1016/j.rser.2016.12.104>

- [4] J. Krause, C. Thiel, D. Tsokolis, Z. Samaras, C. Rota, A. Ward, P. Prenninger, T. Coosemans, S. Neugebauer, and W. Verhoeve, "EU road vehicle energy consumption and CO₂ emissions by 2050—Expert-based scenarios," *Energy Policy*, vol. 138, p. 111224, 2020. doi: <https://doi.org/10.1016/j.enpol.2019.111224>
- [5] C. Fernández-Dacosta, L. Shen, W. Schakel, A. Ramirez, and G. J. Kramer, "Potential and challenges of low-carbon energy options: Comparative assessment of alternative fuels for the transport sector," *Applied Energy*, vol. 236, pp. 590–606, 2019. doi: <https://doi.org/10.1016/j.apenergy.2018.11.055>
- [6] B. Djordjević and E. Krmac, "Evaluation of energy-environment efficiency of European transport sectors: non-radial DEA and TOPSIS approach," *Energies*, vol. 12, no. 15, p. 2907, 2019. doi: <https://doi.org/10.3390/en12152907>
- [7] A. Safari, N. Das, O. Langhelle, J. Roy, and M. Assadi, "Natural gas: A transition fuel for sustainable energy system transformation?" *Energy Science & Engineering*, vol. 7, no. 4, pp. 1075–1094, 2019. doi: <https://doi.org/10.1002/ese3.380>
- [8] D. Chiaramonti and K. Maniatis, "Security of supply, strategic storage and Covid19: Which lessons learnt for renewable and recycled carbon fuels, and their future role in decarbonizing transport?" *Applied Energy*, vol. 271, p. 115216, 2020. doi: <https://doi.org/10.1016/j.apenergy.2020.115216>
- [9] S. Pfoser, O. Schauer, and Y. Costa, "Acceptance of LNG as an alternative fuel: Determinants and policy implications," *Energy Policy*, vol. 120, pp. 259–267, 2018. doi: <https://doi.org/10.1016/j.enpol.2018.05.046>
- [10] Z. Navas-Anguita, D. García-Gusano, and D. Iribarren, "A review of techno-economic data for road transportation fuels," *Renewable and Sustainable Energy Reviews*, vol. 112, pp. 11–26, 2019. doi: <https://doi.org/10.1016/j.rser.2019.05.041>
- [11] J. Wątróbski, A. Bączkiewicz, E. Ziemia, and W. Sałabun, "Sustainable cities and communities assessment using the DARIA-TOPSIS method," *Sustainable Cities and Society*, p. 103926, 2022. doi: <https://doi.org/10.1016/j.scs.2022.103926>
- [12] J. Wątróbski, A. Bączkiewicz, and W. Sałabun, "pyrepmcda-Reference objects based MCDA software package," *SoftwareX*, vol. 19, p. 101107, 2022. doi: <https://doi.org/10.1016/j.softx.2022.101107>
- [13] Y. A. Solangi, C. Longsheng, and S. A. A. Shah, "Assessing and overcoming the renewable energy barriers for sustainable development in Pakistan: An integrated AHP and fuzzy TOPSIS approach," *Renewable Energy*, vol. 173, pp. 209–222, 2021. doi: <https://doi.org/10.1016/j.renene.2021.03.141>
- [14] G. D. Oliveira and L. C. Dias, "The potential learning effect of a MCDA approach on consumer preferences for alternative fuel vehicles," *Annals of Operations Research*, vol. 293, no. 2, pp. 767–787, 2020. doi: <https://doi.org/10.1007/s10479-020-03584-x>
- [15] W. Chmielarz and M. Zborowski, "On the Assessment of e-Banking Websites Supporting Sustainable Development Goals," *Energies*, vol. 15, no. 1, p. 378, 2022. doi: <https://doi.org/10.3390/en15010378>
- [16] P. Ziemia, "Selection of Electric Vehicles for the Needs of Sustainable Transport under Conditions of Uncertainty—A Comparative Study on Fuzzy MCDA Methods," *Energies*, vol. 14, no. 22, p. 7786, 2021. doi: <https://doi.org/10.3390/en14227786>
- [17] A. Romero-Ania, L. Rivero Gutiérrez, and M. A. De Vicente Oliva, "Multiple criteria decision analysis of sustainable urban public transport systems," *Mathematics*, vol. 9, no. 16, p. 1844, 2021. doi: <https://doi.org/10.3390/math9161844>
- [18] L. Rivero Gutiérrez, M. A. De Vicente Oliva, and A. Romero-Ania, "Economic, Ecological and Social Analysis Based on DEA and MCDA for the Management of the Madrid Urban Public Transportation System," *Mathematics*, vol. 10, no. 2, p. 172, 2022. doi: <https://doi.org/10.3390/math10020172>
- [19] J. L. Osorio-Tejada, E. Llera-Sastresa, and S. Scarpellini, "A multi-criteria sustainability assessment for biodiesel and liquefied natural gas as alternative fuels in transport systems," *Journal of Natural Gas Science and Engineering*, vol. 42, pp. 169–186, 2017. doi: <https://doi.org/10.1016/j.jngse.2017.02.046>
- [20] M. A. Hatefi, "A Multi-Criteria Decision Analysis Model on the Fuels for Public Transport, with the Use of Hybrid ROC-ARAS Method," *Petroleum Business Review*, vol. 2, no. 1, pp. 45–55, 2018. doi: <https://dx.doi.org/10.22050/pbr.2018.77848>
- [21] J. Wątróbski, J. Jankowski, and Z. Piotrowski, "The selection of multicriteria method based on unstructured decision problem description," in *International Conference on Computational Collective Intelligence*. Springer, 2014. doi: https://doi.org/10.1007/978-3-319-11289-3_46 pp. 454–465.
- [22] J. Jankowski, K. Kolomvatsos, P. Kazienko, and J. Wątróbski, "Fuzzy modeling of user behaviors and virtual goods purchases in social networking platforms," *Journal of Universal Computer Science*, vol. 22, no. 3, pp. 416–437, 2016. doi: <http://dx.doi.org/10.3217/jucs-022-03-0416>
- [23] J. Dezert, A. Tchamova, D. Han, and J.-M. Tacnet, "The spotis rank reversal free method for multi-criteria decision-making support," in *2020 IEEE 23rd International Conference on Information Fusion (FUSION)*. IEEE, 2020. doi: <https://doi.org/10.23919/FUSION45008.2020.9190347> pp. 1–8.
- [24] J. Jankowski, W. Sałabun, and J. Wątróbski, "Identification of a multi-criteria assessment model of relation between editorial and commercial content in web systems," in *Multimedia and Network Information Systems*. Springer, 2017, pp. 295–305.
- [25] S. Goswami and S. Mitra, "Selecting the best mobile model by applying AHP-COPRAS and AHP-ARAS decision making methodology," *International Journal of Data and Network Science*, vol. 4, no. 1, pp. 27–42, 2020. doi: <http://dx.doi.org/10.5267/j.ijdns.2019.8.004>
- [26] B. Bera, P. K. Shit, N. Sengupta, S. Saha, and S. Bhattacharjee, "Susceptibility of deforestation hotspots in Terai-Dooars belt of Himalayan Foothills: A comparative analysis of VIKOR and TOPSIS models," *Journal of King Saud University-Computer and Information Sciences*, 2021. doi: <https://doi.org/10.1016/j.jksuci.2021.10.005>
- [27] J. Martins and F. Brito, "Alternative fuels for internal combustion engines," *Energies*, vol. 13, no. 16, p. 4086, 2020. doi: <https://doi.org/10.3390/en13164086>
- [28] M. Sajjad, W. Sałabun, S. Faizi, M. Ismail, and J. Wątróbski, "Statistical and analytical approach of multi-criteria group decision-making based on the correlation coefficient under intuitionistic 2-tuple fuzzy linguistic environment," *Expert Systems with Applications*, vol. 193, p. 116341, 2022. doi: <https://doi.org/10.1016/j.eswa.2021.116341>
- [29] Eurostat, *Final energy consumption in road transport by type of fuel*, 2022. [Online]. Available: <https://ec.europa.eu/eurostat/databrowser/view/ten00127/default/table?lang=en>
- [30] J. Wątróbski, J. Jankowski, P. Ziemia, A. Karczmarczyk, and M. Ziolo, "Generalised framework for multi-criteria method selection," *Omega*, vol. 86, pp. 107–124, 2019. doi: <https://doi.org/10.1016/j.omega.2018.07.004>
- [31] E. Ziemia, "The contribution of ICT adoption to sustainability: Households' perspective," *Information Technology & People*, vol. 32, no. 3, pp. 731–753, 2019. doi: <https://doi.org/10.1108/ITP-02-2018-0090>
- [32] A. Bączkiewicz, B. Kizielewicz, A. Shekhovtsov, J. Wątróbski, and W. Sałabun, "Methodical Aspects of MCDM Based E-Commerce Recommender System," *Journal of Theoretical and Applied Electronic Commerce Research*, vol. 16, no. 6, pp. 2192–2229, 2021. doi: <https://doi.org/10.3390/jtaer16060122>
- [33] E. Ziemia, "The contribution of ICT adoption to the sustainable information society," *Journal of Computer Information Systems*, vol. 59, no. 2, pp. 116–126, 2019. doi: <https://doi.org/10.1080/08874417.2017.1312635>
- [34] N. C. Onat, "How to compare sustainability impacts of alternative fuel Vehicles?" *Transportation Research Part D: Transport and Environment*, vol. 102, p. 103129, 2022. doi: <https://doi.org/10.1016/j.trd.2021.103129>
- [35] A. Lindfors, R. Feiz, M. Eklund, and J. Ammenberg, "Assessing the potential, performance and feasibility of urban solutions: methodological considerations and learnings from biogas solutions," *Sustainability*, vol. 11, no. 14, p. 3756, 2019. doi: <https://doi.org/10.3390/su11143756>
- [36] S. Dahlgren and J. Ammenberg, "Sustainability Assessment of Public Transport, Part II—Applying a Multi-Criteria Assessment Method to Compare Different Bus Technologies," *Sustainability*, vol. 13, no. 3, p. 1273, 2021. doi: <https://doi.org/10.3390/su13031273>
- [37] J. Ammenberg and S. Dahlgren, "Sustainability Assessment of Public Transport, Part I—A Multi-Criteria Assessment Method to Compare Different Bus Technologies," *Sustainability*, vol. 13, no. 2, p. 825, 2021. doi: <https://doi.org/10.3390/su13020825>
- [38] W. Chmielarz and M. Zborowski, "Towards sustainability in E-banking website assessment methods," *Sustainability*, vol. 12, no. 17, p. 7000, 2020. doi: <https://doi.org/10.3390/su12177000>

Critical Success Factors for Adopting Electronic Document Management Systems in Government Units

Ewa Ziemba
University of Economics
in Katowice
1 Maja 50, 40-287 Katowice, Poland
ewa.ziemba@ue.katowice.pl

Tomasz Papaj
University of Economics
in Katowice
1 Maja 50, 40-287 Katowice, Poland
tomasz.papaj@ue.katowice.pl

Danuta Descours
Marshal's Office of the Silesian
Voivodeship
Powstańców 34, 40-037 Katowice,
Poland
descoursd@slaskie.pl

Abstract—This study aims at proposing the framework of critical success factors (CSFs) for the adoption of an electronic document management system (EDMS) in government units and at identifying such factors for local government in Poland. The study was based on a literature review, interviews with field experts, and a questionnaire survey. The results suggest a framework of 23 factors that were considered prerequisites for successful EDMS adoption. The factors are grouped into four categories to reflect economic, organizational, technological, and legal issues. Furthermore, nine CSFs for EDMS adoption in Polish local government units were identified. They include legislation imposing the obligation on EDMS adoption, technological readiness, quality of back-office software and its integration with front-office and ERP software, employees' awareness of EDMS adoption, front- and back-office software functionality and scope of its adoption, and information security. It is worth noticing that there are no economic factors in the set of the nine CSFs.

I. INTRODUCTION

In recent years, government units in Poland have been and still are subject to significant organizational and technological changes, like the public administration in the world [1]–[6]. The main goal of the changes is to improve the quality of government services and to adapt them to the constantly increasing demands of customers, especially citizens and businesses [7]–[9]. Information-communication technologies (ICTs) in public administration are necessary to provide an electronic communication contact with the customer, minimizing the number or even eliminating the need for customer visits to a government unit [10]. The implementation of appropriate technological solutions related to the introduction of organizational and legal changes allows government clients to safely submit documents, applications, and letters to government units, and receive responses through the same channel [11].

To meet such challenges, government units adopt ICT to improve customer relations and support internal processes related to document collection and management [12]–[15]. Solutions of this type are called electronic document management systems (EDMS). Generally, EDMS may be defined as “an umbrella concept covering various technologies including document imaging, document retrieval, reporting,

character recognition, document management, workflow, form processing, content management, digital signature management, storage and archiving technologies, business process management, and collaboration.” [16]

In this paper, EDMS is specified as an approach in the electronic management of documents across document entire lifecycle, i.e., ranging from submitting a document by a client to a government unit electronically to a client receiving a response from a government unit electronically. In this approach, paper circulation at each step of the document lifecycle is eliminated. In general, EDMS requires two kinds of software, i.e., front- and back-office software. Front-office software is available for government clients and provides them with electronic forms through which they can submit a relevant document (application, letter) to government units and receive a response.¹ On the other hand, back-office software is available to employees of government units and is used to manage these documents (applications, letters) in government units, i.e., register documents received from clients, manage processes related to these documents, and prepare and send responses to clients.²

The provisions of the law in force in Poland allow government units to resign from paper documents in whole or in part, depending on the type of public service provided [17]. Nonetheless, despite this possibility, a small number of government units decide to switch to EDMS³ as the so-called basic system [18], i.e., one in which all processes and activities related to the provision of individual public services and the handling of related documents take place electronically without the use of paper documents.

However, the studies conducted so far have not analyzed the reasons why processes in government units are still carried out in the traditional – paper system, even though they have the appropriate software. The factors determining EDMS adoption as the basic system were also not analyzed, and no recommendations were given to improve this state of affairs. Such knowledge gaps represent a research gap. Therefore, it allowed us to formulate the research objectives, which are: proposing the framework of critical success factors (CSFs) for the adoption of EDMS in government units and identifying such factors for local government in Poland.

Consistent with the purpose, the remainder of the paper is organized as follows. Section 2 reviews the current research

¹ In Poland, the front-office software that the client uses to submit letters, applications, and documents to government units is called ESP. On the other hand, ePUAP is an example of front-office software that allows clients to submit letters, applications, and documents and receive responses.

² In Poland, such software is called eSOD.

³ In Poland is called EZD.

on CSFs for EDMS adoption in organizations. Section 3 describes the research methodology and the data set used for the empirical work. Based on these data, Section 4 presents the results, including a framework of CSFs for EDMS adoption within government units and an assessment of the factors. Section 5 provides the study's contributions, implications, and limitations as well as considerations for future investigative work.

II. THEORETICAL BACKGROUND

The methodological basis for identifying the determinants of EDMS adoption is provided by the theory of CSFs [19]. According to this theory, CSFs are those areas and operations which determine success in a project and lead to obtaining desired objectives. Pursuant to this theory, it is, therefore, necessary to identify the most significant areas in electronic document management on which activities must be focused to achieve the assumed goal – success in adopting EDMS. This goal is directly related to the elimination of paper documents and the paper flow of documents.

The literature presents the results of research on the determinants of EDMS systems, however, they mainly concern business organizations, and only few relate to public organizations [16], [20].

Smyth pointed out two CSFs for the adoption of EDMS in public organizations, i.e., an information-sharing culture and senior management support [21].

Johnston and Bowen noted that EDMS adoption in organizations requires the engagement of policymakers and EDMS users and EDMS integration with the organization's processes. They also emphasized the necessity to educate, advise, and support EDMS users [22].

Hjelt identified three CSFs for EDMS in business organizations, i.e., EDMS and information quality, support quality (training, guidelines), and technological infrastructure [23].

Dyczkowski, in his research on improving the efficiency of IT projects in public management in Poland, listed among the success factors of IT projects, which is undoubtedly EDMS adoption: involvement of EDMS users, management support, experienced project manager, competent project team, efficient communication in the project team, clear business goals of the project, defined EDMS user requirements, ICT infrastructure standards and linking EDMS adoption goals with the organization's business strategy and with the personal goals of team members (motivation) [24].

Based on the study of Australian local and state government units, Nquyen indicated nine success factors for EDMS in the public sector: adequate and ongoing training and support, top management support, staff recordkeeping awareness and practice, excellent strategies of change management, good project management, motivated great implementation team, clear business vision and plan, system performance monitoring and management, well-prepared file plan [25].

McLeod et al. examined the literature on electronic documents and record management from 1996 to early 2009

[20]. They uncovered 44 case studies of EDMS adoption, of which 16 were related to the public sector. The following CSFs for EDMS projects were indicated: aligning projects with business objectives, chief executive commitment, and support of officers, involving employees at all levels within the organization including external stakeholders. According to the authors, communication, piloting and testing, change management, training, and support for users, policies, and guidelines are as critical as good planning and project management, and the existence or development of necessary "infrastructures" and demonstrating benefits.

Based on insights gleaned from a case analysis of practices and experiences of a local government in the UK, that has implemented an EDMS, Jones identified a set of lessons for EDMS adoption which include feasibility study, senior executive commitment, aligned business strategy, project management, improvements to user ownership, training, system utilization, information management processes, printing strategy, and post-implementation review [26].

Based on the existing studies and content analysis approach, Abdulkadhim et al. indicated 14 common factors related to the adoption of EDMS divided into three types, i.e., organizational, technical, and users [27]. Organizational factors included top management support, budget/cost, strategic plan, legislation environment, and collaboration. Technical factors embraced ICT infrastructure, EDMS implementation team, security and privacy/trust, user requirements, data quality, and system integration. Users factors were related to awareness, staff training, and resistance to change.

Alshibly et al. composed a list of 37 factors that were considered prerequisites for successful EDMS adoption. Then these 37 factors were grouped into six categories, i.e., technological readiness, top management support, training and involvement, resource availability, system-related factors, work environment, and culture [16]. Through a questionnaire survey and factor analysis, the authors confirmed that the factor group "system-related factors" was deemed the most important of all for successful EDMS implementation, followed by "top management support," "resource availability," "training and involvement," "technological readiness," and "work environment and culture."

Aziz et al. identified ten factors based on a literature review, UTAUT (Unified Theory of Acceptance and Use of Technology) and ISSM (Delone-McLean Information System Success Model), as well as experts' opinions. The factors included performance expectancy, effort expectancy, social influence, facilitating condition, system quality, information quality, service quality, the perceived value of records, policy, and security [28].

In summary, the most frequently mentioned success factors are top management support, internal communication, change management, a competent and properly selected project team, training for employees, as well as appropriate hardware and software infrastructure.

III. RESEARCH METHODOLOGY

A multi-step approach was applied in our research methodology:

1. Reviewing the literature. The general purpose of this step was to critically synthesize and appraise the current state of knowledge related to CSFs for EDMS adoption. The search for the appropriate literature began with five bibliographic databases, that is Ebsco, Science Direct, Web of Science, and Scopus. This was achieved by developing a relevant set of keywords and phrases such as “critical success factors,” “CSFs,” “electronic document management” “electronic government,” “success factors,” and “success” in all possible permutations and combinations (taking into consideration the logical AND, and OR as appropriate) and conducting a corresponding search. In addition, Google Scholar was searched to find some relevant literature, especially describing Polish experiences in EDMS adoption for governments. The set of CSFs for EDMS identified in the literature is included in Table I.
2. Defining and verifying the prototype framework of CSFs. This step required a combination of theoretical knowledge and practical experience. Only theoretical knowledge based on the literature review and practical experience based on working in practice can provide insights to indicate meaningful factors influencing e-government. Therefore, the factors indicated based on the literature were analyzed in the context of Polish government circumstances and the prototype CSFs for EDMS adoption were indicated on the basis of action research. The action research means the very close longstanding collaboration of the research team members with public government units in Silesia Province (Poland) which plan, implement, and use EDMS. Then, the 23 prototype CSFs were examined and verified through the means of interviews with field experts, i.e., employees of local and state governments who are responsible for EDMS adoption.
3. Creating the final framework of CSFs. At this stage, a survey questionnaire covering 23 CSFs was prepared. The survey question was: On a scale of 1-5 state to what extent do you agree that the following factors influence the adoption of EDMS in government units? A Likert scale was used to evaluate the strength of the influence of particular factors on EDMS adoption, which represented: 1 – disagree strongly, 2 – disagree, 3 – neither agree nor disagree, 4 – agree, 5 – agree strongly, respectively. Then, the pilot study was conducted in which 28 experts participated. The selection of experts was made in such a manner as to combine the knowledge and experience of scholars, researchers, and practitioners. The experts were employees of local and state government (25) who are responsible for ICTs and e-government adoption, and professors of Polish universities (3) who conduct studies and empirical research on e-government. The pilot study was carried out between May 8, 2017 and June 9, 2017. The variability and reliability analyses (the value of Cronbach’s alpha coefficient was 0.91) of data collected proved the internal consistency of factors and underpinned the reasoning behind the decision to conduct further study of the 23 CSF framework. Additionally, at this step, some

- experts proposed minor changes in the prototype factors related to confusing or incomprehensible statements. It allowed us to elaborate on the final framework of CSFs (Table I) and the final version of the survey questionnaire.
4. Assessing 23 CSFs proposed and identifying CSFs for government units in Silesia Province. Having applied the Computer Assisted Web Interview, the survey questionnaire was uploaded to the website and submitted to all 185 government units in Silesia Province. The respondents were advised that their participation in completing the survey was voluntary. At the same time, they have been assured anonymity and guaranteed that their responses would be kept confidential. The data were collected between July 11, 2017 and September 19, 2017. After screening the responses and excluding outliers, there was a final sample of 110 usable, correct, and complete responses. It means that 60% of all government units from Silesia Province completed their responses fairly, in all respects. The sample ensured that the error margin for the 95% confidence interval was 5%. Government units varied in their types and the number of employees. The data were stored in Microsoft Excel format. Using the Statistica package and Microsoft Excel, the data were analyzed. The descriptive statistical analysis was employed to identify CSFs (Table I). The following statistics were calculated: min, max, mean, median (MDN), standard deviation (SD), and coefficient of variation (CV).

IV. FINDINGS AND DISCUSSION

Based on the literature review (step 1 of the research process), the interviews with experts (step 2 of the research process), and the pilot survey questionnaire (step 3 of the research process), a framework of 23 economic, organizational, technological, and legal CSFs for the adoption of EDMS in government units were identified (Table I).

Table I presents the detailed descriptive analysis of all 23 CSFs examined. Out of these factors, nine factors were identified that obtained the highest values of the arithmetic mean (above 4.0) and the median (equal to 4.0) in the study. These factors have been seen as critical for EDMS adoption as the base system. Using the Pareto [29] principle as critical among 23 factors, it was necessary to focus on five of them, those that determine EDMS adoption to the greatest extent. However, in the end, this number of factors was extended to nine, which was due to the slight differences in the values of their arithmetic means and medians, and expert opinions.

To sum up, the results of statistical analyses allowed us to recommend nine CSFs for EDMS adoption in Polish government units. These are: Legislation imposing the obligation on EDMS adoption (X23); Back-office software functionality and scope of its adoption (X19); Employees’ awareness of EDMS adoption (X4); Quality of back-office software (X14); Integration of back-office and ERP software (X13); Technological readiness (X11); Integration of front-and back-office software (X12); Information security (X18); Front-office software functionality and scope of its adoption (X20).

TABLE I.
A FRAMEWORK OF CSFs FOR THE ADOPTION OF EDMS IN GOVERNMENT UNITS AND DESCRIPTIVE STATISTICS OF 23 CSFs

No	Description	Type*	Min	Max	Mean	MDN	SD	CV (%)
X23	Legislation imposing the obligation on EDMS adoption	L	2	5	4.19	4	0.851	20.31
X19	Back-office software functionality and scope of its adoption	T/O	2	5	4.15	4	0.768	18.50
X4	Employees' awareness of EDMS adoption	O	1	5	4.13	4	1.015	24.58
X14	Quality of back-office software	T	1	5	4.12	4	0.906	22.00
X13	Integration of back-office and ERP (Enterprise Resource Planning) software	T	1	5	4.10	4	0.928	22.64
X11	Technological readiness	T	1	5	4.06	4	0.998	24.56
X12	Integration of front- and back-office software	T	1	5	4.05	4	0.956	23.58
X18	Information security	T/O	1	5	4.03	4	0.981	24.36
X20	Front-office software functionality and scope of its adoption	T/O	2	5	4.01	4	0.862	21.50
X7	Information culture	O	1	5	3.96	4	0.877	22.13
X22	Legal regulations, procedures, policies, and guidelines	L	1	5	3.95	4	0.85	21.50
X15	Quality of front-office software	T	1	5	3.94	4	0.979	24.88
X16	Maturity of e-government services	T	1	5	3.92	4	1.006	25.67
X5	Employees' soft competences	O	1	5	3.91	4	0.963	24.64
X21	Integration of solutions at local and national levels, and their interoperability	L/T/O	1	5	3.91	4	1.045	26.74
X2	Expenditure on employees' ICT education and training	E	1	5	3.80	4	0.936	24.65
X9	Top management support	O	1	5	3.80	4	0.865	22.77
X6	Motivated and involved employees	O	1	5	3.69	4	0.983	26.65
X1	Expenditure on ICT	E	1	5	3.68	4	1.156	31.42
X8	Competent great adoption team	O	1	5	3.64	4	1.081	29.74
X3	Demonstrating economic benefits	E	1	5	3.59	4	0.979	27.28
X17	ICT risk management	T	1	5	3.54	4	0.974	27.54
X10	Management concepts adoption	O	1	5	3.08	3	0.858	27.84

* Notes: E – Economic, O – Organizational, T – Technological, L – Legal

It is worth noticing that there are no economic factors in the set of the nine CSFs. In the opinion of the government units examined, technological, organizational, and legal factors are the most important ones. A critical factor for EDMS adoption is a legal factor, i.e., legislation imposing the obligation on EDMS adoption in government units. Introducing mandatory electronic communication stimulates and accelerates front- and back-office software adoption for document management for a broad range of government processes, government services, relations with government clients, and relations between government employees within government units and between government units. Critical factors also include technological factors, i.e., technological readiness, quality of back-office software, and its integration with front-office and ERP software. Such integration is of great importance in improving the organization of work, processes, and document workflow, e.g., by eliminating duality – the need to conduct double document workflows (paper and electronic). The challenges, benefits, and risks of EDMS adoption also have a critical impact on EDMS adoption. The CSFs for EDMS consist of organizational and technological factors, including organizational solutions and methods, as well as technological issues. One of them is front- and back-office software functionality and the scope of its adoption, i.e., the

usage of EDMS for managing various documents and providing various government services for government clients. Information security, which includes technological solutions, organizational procedures, and legal regulations, is also critical for EDMS adoption.

V. CONCLUSIONS

Generally speaking, the adoption of EDMS poses a challenge and thus is an interesting subject of research. This research puts an effort to make some contribution to the development of studies on EDMS, especially on their successful adoption in government units. It explores the EDMS concept, indicates CSFs for EDMS based on the literature review, and identifies a comprehensive set of CSFs based on action research and expert interviews. Finally, it proposes the framework of CSFs for EDMS adoption. The research findings showed technological, organizational, and legal factors matter in accelerating the ability and willingness of government units to adopt EDMS successfully.

The CSFs framework proposed can be useful for transition, emerging, and developing economies, especially in Central and East Europe. Government practitioners could find answers to an important question: which areas and activities of government units should be a primary focus for achieving

the most satisfying results of transforming traditional document management to electronic document management. This research suggests important issues for programming, building, and developing EDMS.

The framework of critical success factors for EDMS adoption shown in this research should be explored in greater depth. By focusing on longitudinal research and expanding the number of government units examined from various countries, the authors hope to thoroughly verify this framework. Furthermore, there is also a need to conduct more in-depth research on EDMS, especially in: (1) exploring “best practices” to be used to successfully adopt EDMS, (2) investigating the “demand-side” of EDMS from the viewpoint of government units clients, i.e., citizens and businesses view, and (3) identifying strengths, weaknesses, opportunities, and threats of EDMS in government units. Those will be considered as future work.

VI. REFERENCES

- [1] D. MacLean, and R. Titah, “A Systematic Literature Review of Empirical Research on the Impacts of e-Government: A Public Value Perspective,” *Public Administration Review*, vol. 82, no 1, pp. 23-38, 2022. <https://doi.org/10.1111/puar.13413>
- [2] N. Darmawan, “A Bibliometric Analysis of E-Government Research,” *Library Philosophy and Practice (e-journal)*. 5861, 2021. <https://digitalcommons.unl.edu/libphilprac/5861>
- [3] J. D. Twizeyimana, and A. Andersson, “The public value of E-Government – A literature review,” *Government Information Quarterly*, vol. 36, no 2, pp. 167-178, 2019. <https://doi.org/10.1016/j.giq.2019.01.001>
- [4] J. M. Sánchez-Torres, and I. Miles, “The role of future-oriented technology analysis in e-Government: a systematic review,” *European Journal of Futures Research*, vol. 5, no 15, pp. 1-18, 2017. <https://doi.org/10.1007/s40309-017-0131-7>
- [5] E. Ziemba, “The Contribution of ICT Adoption by Local Governments to Sustainability—Empirical Evidence from Poland,” *Information Systems Management*, vol. 38, no 2, pp. 116–134, 2020. <https://doi.org/10.1080/10580530.2020.1738600>
- [6] E. Ziemba, “The ICT adoption in government units in the context of the sustainable information society,” in: *Proceedings of the 2018 Federated Conference on Computer Science and Information Systems FedCSIS*, M. Ganzha, L. Maciaszek, and M. Paprzycki, Eds., Adam Mickiewicz University, September 9-12, 2018, pp. 725-733, 2020. <https://doi.org/10.15439/2018F116>
- [7] D. Agostino, I. Saliterer, and I. Steccolini, “Digitalization, accounting and accountability: A literature review and reflections on future research in public services,” *Financial Accountability and Management*, vol. 38, no 2, pp. 152-176, 2022. <https://doi.org/10.1111/faam.12301>
- [8] European Commission, *eGovernment Benchmark 2021: Entering a new digital government era*, Publication Office of the European Union, Luxembourg, 2021.
- [9] I. Lindgren, and G. Jansson, “Electronic Services in the Public Sector: A Conceptual Framework,” *Government Information Quarterly*, 2013, vol. 30, no 2, pp. 163-172, 2013. <https://doi.org/10.1016/j.giq.2012.10.005>
- [10] M. L. Correa Ospina, D. Saxena, and B. H. Díaz Pinzón, “Mechanisms underpinning the usage of e-government services by businesses: A proposal based on previous empirical research,” *JeDEM - eJournal of eDemocracy and Open Government*, vol. 13, no. 2, 2021, Ongoing Papers. <https://doi.org/10.29379/jedem.v13i2.685>
- [11] F. K. Y. Chan, J. Y. L. Thong, S. A. Brown, and Viswanath Venkatesh, “Service Design and Citizen Satisfaction with E-Government Services: A Multidimensional Perspective,” *Public Administration Review*, vol. 81, no 5, pp. 874-894, 2021. <https://doi.org/10.1111/puar.13308>
- [12] A. A. Aziz, Z. M. Yusof, U. A. Mokhtar, and D. I. Jambari, “The implementation guidelines of digital document management system for Malaysia public sector: Expert review,” *International Journal on Advanced Science, Engineering and Information Technology*, vol. 10, no 1, pp.198-204, 2020.
- [13] A. Zuiderwijk, Y.-C. Chen, and F. Salem, “Implications of the use of artificial intelligence in public governance: A systematic literature review and a research agenda,” *Government Information Quarterly*, vol. 38, no 2, 101577, 2021. <https://doi.org/10.1016/j.giq.2021.101577>
- [14] J. Amend, J. Kaiser, L. Uhlig, N. Urbach, and F. Völter, “What Do We Really Need? A Systematic Literature Review of the Requirements for Blockchain-Based E-government Services,” in: *Innovation Through Information Systems. WI 2021. Lecture Notes in Information Systems and Organisation*, F. Ahlemann, R. Schütte, and S. Stieglitz, Eds., Springer, vol 46, 2021. https://doi.org/10.1007/978-3-030-86790-4_27
- [15] C. Fredriksson, F. Mubarak, M. Tuohimaa, and M. Zhan, “Big Data in the Public Sector: A Systematic Literature Review,” *Scandinavian Journal of Public Administration*, vol. 21, no 3, pp. 39-61, 2017.
- [16] H. Alshibly, R. Chiong, and Y. Bao, “Investigating the Critical Success Factors for Implementing Electronic Document Management Systems in Governments: Evidence From Jordan,” *Information Systems Management*, vol. 33, no 4, pp. 287-301, 2016. <https://doi.org/10.1080/10580530.2016.1220213>
- [17] Regulation of the Prime Minister of January 18, 2011, on document management offices, uniform subject file indexes, and instructions on the organization and scope of operation of institutional archives (Journal of Laws 2011 No. 14 item 67).
- [18] Główny Urząd Statystyczny Urząd Statystyczny w Szczecinie, *Spoleczeństwo informacyjne w Polsce w 2020 r.*, Warszawa, Szczecin, pp. 51-53, 2020.
- [19] J. F. Rockart, and C. Bullen, *A Primer on Critical Success Factors*, Center for Information Systems Research Working Paper, 69, Sloan School of Management, MIT, Cambridge, 1981.
- [20] J. McLeod, S. Childs, and R. Hardiman, “Accelerating positive change in electronic records management: Headline findings from a major research project,” *Archives and Manuscripts*, vol. 39, no 2, pp. 66-94, 2011.
- [21] Z. Smyth, “Implementing EDRM: Has it provided the benefits expected?,” *Records Management Journal*, vol. 15, no 3, pp.141-149, 2005. <https://doi.org/10.1108/09565690510632328>
- [22] G. Johnston, and D. Bowen, “The benefits of electronic records management systems: A general review of published and some unpublished cases,” *Records Management Journal*, vol. 15, no 3, pp. 131-145, 2005. <https://doi.org/10.1108/09565690510632319>
- [23] M. Hjelt, *End-user attitudes towards EDM use in project work: a case study of the Kamppi Center Project (Master’s thesis)*, Department of Management and Organization, Swedish School of Economics and Business Administration (HANKEN), Helsinki, pp. 90, 2006.
- [24] M. Dyczkowski, *Wiedza o krytycznych czynnikach sukcesu jako istotny element poprawy efektywności przedsięwzięć informatycznych w sferze zarządzania publicznego*, Akademia Ekonomiczna, Wrocław, 2007.
- [25] L. T. Nguyen, P. M. C. Swatman., B. Fraunholz, and S. Salzman, “EDRMS implementation in the Australian public sector,” in: *Proceedings of the 20th Australasian Conference on Information Systems*, Melbourne, pp. 915-928, 2009.
- [26] S. Jones, “eGovernment Document Management System: A case analysis of risk and reward,” *International Journal of Information Management*, vol. 32, no 4, pp. 396-400, 2012. <http://doi.org/10.1016/j.ijinfomgt.2012.04.002>
- [27] H. Abdulkadhim, M. Bahari, A. Bakri, and H. Hashim, “Exploring the common factors influencing electronic document management systems (EDMS) implementation in government,” *ARPN Journal of Engineering and Applied Sciences*, vol. 10, no 23, pp. 17945-17952, 2015.
- [28] A.A. Aziz, Z.M. Yusof, U.A. Mokhtar, and D.I. Jambari, “A Conceptual Model for Electronic Document and Records Management System Adoption in Malaysian Public Sector,” *International Journal on Advanced Science Engineering and Information Technology*, vol. 8, no 4, pp.1191-1197, 2018. <https://doi.org/10.18517/ijaseit.8.4.6376>
- [29] J. M. Juran, *Quality Control Handbook*, McGraw-Hill, 1951.

28th Conference on Knowledge Acquisition and Management

KNOWLEDGE management is a large multidisciplinary field having its roots in Management and Artificial Intelligence. Activity of an extended organization should be supported by an organized and optimized flow of knowledge to effectively help all participants in their work.

We have the pleasure to invite you to contribute to and to participate in the conference "Knowledge Acquisition and Management". The predecessor of the KAM conference has been organized for the first time in 1992, as a venue for scientists and practitioners to address different aspects of usage of advanced information technologies in management, with focus on intelligent techniques and knowledge management. In 2003 the conference changed somewhat its focus and was organized for the first under its current name. Furthermore, the KAM conference became an international event, with participants from around the world. In 2012 we've joined to Federated Conference on Computer Science and Systems becoming one of the oldest event.

The aim of this event is to create possibility of presenting and discussing approaches, techniques and tools in the knowledge acquisition and other knowledge management areas with focus on contribution of artificial intelligence for improvement of human-machine intelligence and face the challenges of this century. We expect that the conference&workshop will enable exchange of information and experiences, and delve into current trends of methodological, technological and implementation aspects of knowledge management processes.

TOPICS

- Knowledge discovery from databases and data warehouses
- Methods and tools for knowledge acquisition
- New emerging technologies for management
- Organizing the knowledge centers and knowledge distribution
- Knowledge creation and validation
- Knowledge dynamics and machine learning
- Distance learning and knowledge sharing
- Knowledge representation models
- Management of enterprise knowledge versus personal knowledge
- Knowledge managers and workers
- Knowledge coaching and diffusion
- Knowledge engineering and software engineering
- Managerial knowledge evolution with focus on managing of best practice and cooperative activities
- Knowledge grid and social networks

- Knowledge management for design, innovation and eco-innovation process
- Business Intelligence environment for supporting knowledge management
- Knowledge management in virtual advisors and training
- Management of the innovation and eco-innovation process
- Human-machine interfaces and knowledge visualization

TECHNICAL SESSION CHAIRS

- **Hauke, Krzysztof**, Wroclaw University of Economics, Poland
- **Nycz, Malgorzata**, Wroclaw University of Economics, Poland
- **Owoc, Mieczyslaw**, Wroclaw University of Economics, Poland
- **Pondel, Maciej**, Wroclaw University of Economics, Poland

PROGRAM COMMITTEE

- **Andres, Frederic**, National Institute of Informatics, Tokyo, Japan
- **Berka, Petr**, Prague University of Economics and Business, Czech Republic
- **Bodyanskiy, Yevgeniy**, Kharkiv National University of Radio Electronics, NURE, Ukraine
- **Chomiak-Orsa, Iwona**, Wroclaw University of Economics and Business, Poland
- **Christozov, Dimitar**, The American University in Bulgaria
- **Chudán, David**, University of Economics, Prague, Czech Republic
- **Hernes, Marcin**, Wrocław University of Economics and Business, Poland
- **Jan, Vanthienen**, Katholieke Universiteit Leuven, Belgium
- **Kliegr, Tomáš**, Prague University of Economics and Business, Czech Republic
- **Kluza, Krzysztof**, AGH University of Science and Technology, Poland
- **Ligęza, Antoni**, AGH University of Science and Technology, Poland
- **Mercier-Laurent, Eunika**, Jean Moulin Lyon 3 University, France
- **Perechuda, Kazimierz**, Wroclaw University of Economics and Business, Poland

- **Schreurs, Jeanne**, Hasselt University, Belgium
- **Singh, Pradeep**, KIET Group of Institutions, Delhi-NCR, Ghaziabad, U.P., India
- **Singh, Yashwant**, Jaypee University of Information Technology Wanknaghat, India
- **Sobińska, Małgorzata**, Wrocław University of Economics and Business, Poland
- **Stankosky, Michael**, The University of Scranton, USA
- **Tanwar, Sudeep**, Institute of Technology, Department of CE, Nirma University, Ahmedabad (Gujarat), India
- **Tyagi, Sudhanshu**, Thapar Institute of Engineering & Technology, India
- **Vasiliev, Julian**, University of Economics – Varna, Bulgaria
- **Zhu, Yungang**, College of Computer Science and Technology, Jilin University, China

Case Study of Designing Interface of the AGH Students Information Bulletin Work Support System

Natalia Nitarska, Krzysztof Kluza, Piotr Wiśniewski, Mateusz Zaremba, Antoni Ligeza
AGH University of Science and Technology
al. A. Mickiewicza 30, 30-059 Krakow, Poland
E-mail: {nitarska,wpiotr,kluza,mzaremba,ligeza}@agh.edu.pl

Abstract—In this paper, we present a case study research on designing the system interface for handling organisational processes implemented in the editorial office of the AGH-UST Students Information Bulletin. A study of the basic formal and legal sources, such as regulations and statutes of the organisation, was also carried out for the proper identification of goals, responsibilities, and organisational structure. To deepen this knowledge, social research was carried out, which examined how the users use the current solutions and what their needs are related to the designed system. Based on the research results and the formulated conclusions, the diagrams of business processes carried out in the organisation were created. In the discovered process models, we defined reusable sub-processes using Justified Aggregation of Neighboring Activities. Finally, the information architecture of the target system solution, followed by a medium-fidelity interface were designed.

Index Terms—Business Process Management, BPM, BPMN, process modelling, process models, process knowledge acquisition

I. INTRODUCTION

THE TERM “user experience” was first used in 1993 by Don Norman, who defined his role in the team and job position at Apple Computer Inc. (now Apple Inc.) as a “User Experience Architect” [1]. The company was undoubtedly the forerunner of a breakthrough in thinking about electronic tools used today by a broad and diverse audience. A milestone in thinking about the use of computers and the design of their interfaces was the introduction of the Macintosh computer in 1984 [2]. This was an all-in-one computer with a graphics interface, keyboard, and mouse. Although all of these components had been manufactured and used before, the Macintosh forever changed the perception of what a computer could be used for and who could use it. It was not the unique, exploratory technology behind its success but the well-designed user experience associated with it. The combination of a convenient, intuitive design and an understandable graphical interface meant that computers from laboratories and large companies also moved into homes and became a part of everyday life. Almost 30 years later, our knowledge of interface design has grown so much that, in engaging in this field, we are aware that, in a field so close to human beings, we must take into consideration not only design issues but also psychology, sociology, and many other fields – both technical and humanistic.

In this paper, we discuss the problem associated with the lack of a unified, transparent system for handling work

and processes based on the example of the AGH-UST Media Center at AGH University of Science and Technology in Krakow, Poland. These problems have become particularly noticeable at the time of the organisation’s transition to a remote working mode. The example used in this paper is the design of a process handling interface implemented in one of the four independent editorial boards – the AGH Students’ Information Bulletin. The history of this student magazine dates back to about 1988. Since then, the Bulletin’s editorial team has greatly expanded its scope and today, it not only publishes a periodic student magazine but also maintains a website or podcasts.

Problems in managing the editorial board, which is characterised by frequent replacement of members, became significantly apparent after the transition to a remote mode of operation. When new members are introduced to their procedures completely remotely, and it is impossible to show, add, and explain some processes in person, it is easy to see that functional solutions for coordinating work are introducing mistakes to the new users. In addition, the systems and applications that are used in the organisation not only do not fully meet the needs of its members but using them without discerning the working in the organisation in the land-based mode becomes a challenge.

The purpose of this paper is to present the process of designing a user interface for handling the work of the AGH-UST Student Information Bulletin, being able to almost replace the existing currently used toolbox (tools and platforms), both for project coordination and communication among members. Moreover, the implementation of this system could allow for significant improvement and ease of work for regular members, as well as clearly facilitate the implementation of new, constantly emerging issues organisations. The platform would also allow for more efficient documentation of the work of the editorial board.

This paper is organised as follows: Section II presents the basic principles of user interface design, including the related psychological aspects. Section III presents the motivation behind our research and the analysed problem, including interviews with the study participants. The part of our research that covers process discovery was described in Section IV. We present the final design of the user interface in Section V and finally, Section VI summarises our contribution presented in this paper and provides the overview of the future works.

II. USER INTERFACE DESIGN

According to the book *"Human-Computer Interaction"* [3], the term *HCI* has been in common use since the early 1980s, but research in this area began early in the last century. The original focus was the study of human interaction with machine. As the state of the art and computerization advanced, the focus of the research was changed to human-computer interaction. Information technology (hidden today under the acronym *IT*) is another field that has influenced the development of *HCI*. It is for it today that consideration of *HCI* principles is crucial, as an integral part of the design process. The user (understood as both an individual and a team) interacts with the computer to achieve a specific goal.

A. Shneiderman's Eight Golden Rules of Interface design

A precursor to viewing computer systems from the perspective of the user interface is American computer scientist Ben Shneiderman [4]. Shneiderman today is considered to be one of the third "fathers of User Experience Design". His book *"Designing the User Interface"* [5] published in 1987 was the first complete textbook on how to design user interfaces. Most importantly, the book featured the first set of universal principles for designing UI. The Eight Golden Rules of Dialog Design (*"Eight Golden Rules of Dialog Design"*) and the body of Shneiderman's work and research were also the basis for the development of the field of User Experience Design. Thus, Shneiderman's 8 Golden Principles can be cited [6], [5]:

- 1) **Strive for Consistency**. As the author points out – it is the one most often violated, but failure to follow it is also the easiest to fix and avoid. The principle refers to the need to use the same terminology when giving the same information and marking the same actions, but also to use color codes for interactive and fixed elements, the same icons, call-to-actions, or consistent sequences of actions in similar situations. As defined in the Interaction Design Foundation's article on the Golden Rules – "...the user should be able to use knowledge from one action to perform another." [6]. Situations where consistency is extremely difficult or impossible should be kept to a minimum. Adherence to this principle helps users learn to use the system more quickly and in sequence, achieve their intended goals more quickly.
- 2) **Enable Frequent Users to Use Shortcuts**. As the frequency of system use increases, the desire to reduce the amount of interaction time increases. Hidden commands, remembering once entered data, auto-complete data, and all shortcuts are appreciated by regular users. Following this principle allows users to achieve their goals faster and easier.
- 3) **Offer Informative Feedback**. The principle refers to the fact that with every user action, there should be feedback from the system, telling about the success or lack of success of the action. It is important that the information should be given in an accessible way to the user, rather than being, for example, just a message

about the error code that occurred. The information can be modest for less complex actions, such as displaying the requested information when the user clicks on an item, or more extensive for more complex operations, such as a pop-up with a message about why the action failed.

- 4) **Design Dialog to Yield Closure**. The action the user takes should have a clear beginning and end. This means that the user at each step of the process should know at which point they are. An example of a correct fulfillment of this principle would be the display of a thank you message after a purchase has been made in an online store, or the announcement that data has been successfully updated after the user has made changes to the system.
- 5) **Offer Simple Error Handling**. The system should inform the user of errors it makes, their severity, and type, but should also do so in a manner that is as benign as possible. The interface must not make the user feel guilty for making an error. A way to satisfy this principle might be to indicate specific places, such as form fields, that have been filled in incorrectly or not filled in, rather than having the system reject the entire form only after it has been filled in and submitted without indicating where the error occurred.
- 6) **Permit Easy Reversal of Actions**. The principle refers to allowing the user to reverse their own actions. In performing any action, users will make mistakes, and so the key to reducing their frustration when they make them is to ensure that at any time their mistake can be corrected. Shneiderman also emphasises that knowledge of the ease of undoing actions encourages users to explore unexplored options. Undo options should be possible after a single action as well as a sequence of them.
- 7) **Support Internal Locus of Control**. The principle encourages putting the user in the role of initiator of system actions. The user should feel that it is the user who decides how the system functions - after all, the system is a tool in the user's hand, designed to help the user achieve the desired goal. The user should have a sense of complete control over the processes occurring in the virtual space.
- 8) **Reduce Short-Term Memory Load**. The principle refers to the limited capacity of human short-term memory. To the best of today's knowledge, it can hold 7 plus or minus two items. Therefore, the interface should not require the user to remember more information at once. Additionally, the usage of the interface should be based on recognising elements more than recalling them from memory, which generates more stress for users and consumes more time.

All of the above principles provide a basis for further consideration and development of knowledge about user interface design and human-computer interaction.

B. Ten Usability Heuristics

Another important step in the development of the field of user experience design was Jakob Nielsen's formulation of the 10 Usability Heuristics in 1994. It is a set of ten core principles of interaction design. Nielsen's heuristics are still used today and still form the basis of usable systems design. The content of the heuristics has remained unchanged to this day, but the article in which they were presented was enhanced this year with examples and explanations of the [7]. In reference to Shneiderman's golden rules presented earlier, the essential step seems to be Nielsen's addition of two rules:

- 1) **Match between system and the real world.** The principle refers to enabling users to use the knowledge acquired in the real world to function efficiently in the virtual world. We are talking both about using language familiar to the user, instead of industry jargon, and about following concepts and schemes of operation familiar from the real world. The issue of basing digital products on knowledge about the functioning of the real world was later extended by Don Norman.
- 2) **Aesthetic and minimalist design.** This principle pioneered the perception of aesthetics as an important element of an interface, affecting its perception, and, more importantly, the comfort and effectiveness of its use. Nielsen also cautions against placing redundant information in the interface that is merely decorative or noise. Any unnecessary information distracts from the vital information.

C. Basic psychological concepts related to design

A significant influence on knowledge development in the field of user experience design was Don Norman's book *"The Design of Everyday Things"*, first published in 2002 [8]. It is from it that further concepts that are the basis of today's knowledge in this field were drawn. As Don Norman writes, *"For a product to have transparency, its designers must correctly apply five basic psychological concepts [...], namely affordances, signifiers, constraints, mappings, and feedback. Nevertheless, there is a sixth principle, perhaps the most important of all: the conceptual model of the system"* [9]. Thus, definitions of the basic concepts can be quoted after Norman:

- 1) **Affordances** – is the ability to perform a particular action that an object manifests. Whether an object manifests a given affordance is also dependent on the person who interacts with it. For an adult, a staircase manifests the affordance of ascending or descending it, and from this follows its function – to enable upward or downward movement. For a child who cannot move up or down stairs, stairs will not manifest this affordance, as so they will not serve the same function either. Affordances allow us to identify the function of objects without having to use labels or put additional information on them. The opposite concept to affordances is anti-affordance [10] – the perceived inability

to take a particular action. Importantly, objects should clearly manifest their affordances and anti-affordance so that the user can easily identify them. The problem begins when an object manifests an affordance that is not related to its actual function. This is, for example, a doorknob that manifests an affordance for pressing it, but under pressure, it does not fall at all and does not cause the door to open.

- 2) **Signifiers** – is a specification of how or where an action is performed. Markers can be placed on an object intentionally or be found on it by accident. Markers are often confused with affordances. The concepts are not the same; however, the boundary between the two has also been drawn by Norman – *"Affordances specify what actions are possible. Markers tell you where to perform them. One and the other are necessary"* [9]. Thus, we can call a marker a button in the interface of a system dedicated to cell phones. This marker uses the affordance of touching the screen, which the cell phone had much earlier than the system in it.
- 3) **Constraints** – Constraints (like the knowledge that the user has) can be divided into those that occur in the world and those that occur in the mind. Both types accompany users when using both digital and physical products, and so they should be considered during design:
 - **Boundaries in the world** are those that do not provide a physical way to perform a particular action. This is, for example, the lack of a handle or wheels on a heavy object, the lack of a handle on a door, or the lack of finger holes in a bowling ball,
 - **Boundaries in mind** are related to cultural code, among other things. These are activities that the user must have learned not to do, but the design of the product or system allows for it. It is, for example, the fact that we typically use a computer mouse using our hands, although its design does not preclude using it with our feet or other body parts.
- 4) **Mappings** – (in user experience design) is a way of mapping/showing the relationship between controls and controls. The more natural the mapping, the better the functionality. Norman also distinguished three levels of mapping:
 - **Best mapping** – controls placed on elements controlled.
 - **Second best mapping** – controls placed closest to the controlled elements.
 - **Third of the best mappings** – controls are placed in the same configuration as the controls.
- 5) **Feedback** – this issue has already been directly addressed by one of Goldman's Golden Rules, but has been significantly developed by Norman in *"Design Everyday"* [9]. Among other things, he pointed out

that the type of feedback should be appropriate to the action. Feedback that is too pushy can distract the user and introduce unnecessary chaos. Another mistake to avoid is too much feedback, which is only meant to be part of the process the user goes through, not the backbone of it. In addition to this, the time in which the feedback is given to the user is important. Even a slight delay can cause anxiety, which should be avoided.

- 6) **System conceptual model** – according to an article on mental models on the NNgroup website – “*The conceptual (mental) model of the system is everything the user believes about the system*” [11]. The definition has two important aspects:
- **The conceptual model is based on the user’s knowledge or belief, not on the actual operation and building of the system.** For best results, the conceptual model should be as close as possible to the actual operation of the product.
 - **Every user of the system has their own, different conceptual model of the system.** Consequently, the conceptual model of the designer and the user also differs significantly. The designer’s model is usually based on the greater knowledge of the product that he has. His task, however, is to create the design in such a way that the users’ conceptual models are as close as possible to the actual operation of the system.

All of the above principles and concepts have outlined the theoretical foundations of today’s approach to user experience design, inextricably linked to user interface design.

III. MOTIVATION AND RESEARCH PROBLEM

The main goal of the undertaken activities is to create an interface for the editorial office of the AGH Student Bulletin. In order to achieve the intended purpose, it was necessary to recognise in detail the scope of the needs of the editorial office members. The purpose of the research part of this study was therefore to explore the processes occurring in the editorial office and to create models depicting the tasks and roles of all its departments and external bodies.

One of a major challenges in building complex process models is the identification of sub-processes [12]. The Bulletin is composed of three main departments: journalistic, graphic and promotion and cooperation. Most of the processes are concentrated within them. The projects carried out by the editorial office, however, link the work of all its departments and were therefore treated as sub-processes of the core workflows.

The research included: in part one, an analysis of communication channels and organisational documents provided by the editorial office, and in part two, in-depth individual interviews [13]).

A schematic of the organisational structure of the Media Center is shown in Figure 1, which distinguishes the governing bodies of the Bulletin’s editorial office and those collaborating with it in the processes carried out. The research group consisted of the Editor-in-Chief, his Deputy, and the

heads of the following departments: journalism, graphics, and promotion and cooperation.

A. Interview Structure

The interviews were semi-structured, with detailed questions subject to modification during the interviews. Accordingly, a preliminary scenario was created before the research began.

1) General questions

- How long have you been a member of the Bulletin editorial board?
- How long have you been in your role?
- How many people does the department you manage include?
- How many of them are people who joined you in this semester?
- How many of these people have you worked with for more than a year?

2) Questions about completed projects

- Which projects does your department manage?
- In which other editorial projects are you involved?

3) Identification of responsibilities

- What responsibilities do department members have?
- What are the responsibilities of the person managing the department?
- With what frequency should they be performed?

4) Definition of tools

- What tools (online and offline) do you use in carrying out your responsibilities?
- Can you show me these tools (via the screen share option)?

5) Process discovery

- Imagine that I am a new person in your editorial office. Can you walk me, with the help of screen sharing, through the processes of performing the most important duties in your editorial office/department?

B. Related Works

The main topic discussed in this paper is the user interface design for a work support system. The main principles of building collaborative online applications have been widely discussed since the early 2000s [14]. However, in recent years, one can observe rapid technological developments in the area of Human-Computer Interaction [15] that let the researchers and IT architects look for new ways to improve the overall user experience. In our work, we also mention the role of social media which are increasingly more present as a collaboration tool in companies and organisations [16].

To define use cases for the discussed application, we based our research on Business Process Model and Notation (BPMN), which is a common standard for representing application flows [17]. Although there exist methods to generate

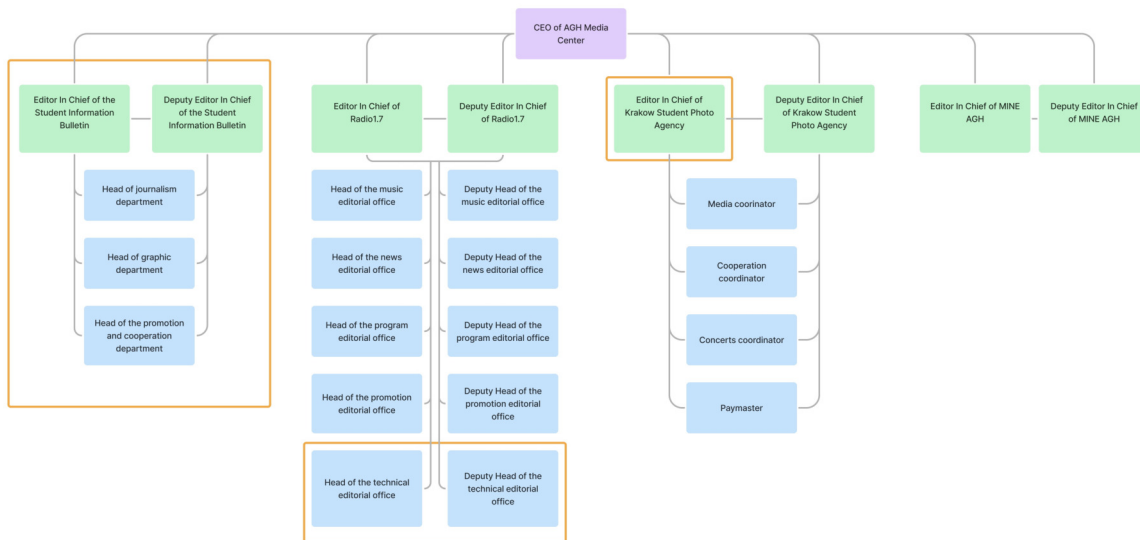


Figure 1. The organisational structure of the AGH-UST Media Center

graphical user interface prototypes based on process models [18], [19], due to large process complexity we decided to perform this action manually.

As it has already been mentioned in this section, one of the significant steps leading to an interface design is the identification and modelling of processes within the organisation. Process discovery can be done using one of the automated process mining techniques [20], one of the common approaches being identifying patterns based on log files [21]. Since the editorial board of the Students Information Bulletin does not use any integrated IT system, automated process mining has been excluded. In such a case, one of the possible approaches is the visual identification of business process elements [22]. In our case, we based our process discovery on user interviews and manual document analysis, analogically to our previous work, where business processes of a training company were identified and discussed [23].

IV. BUSINESS PROCESSES OVERVIEW

This section presents the processes performed in the editorial office, as identified through research. The processes were modelled using BPMN 2.0 notation and described.

The basic structure of the newsletter consists of three departments: journalism, graphic design, and promotion and collaboration. The entire editorial board includes more than thirty members and interns. Each of the departments is run by one managing person. The activities of the entire editorial office are supervised by the Editor-in-Chief and his Deputy.

The main result of the editorial staff’s work is the periodic publication of a magazine – a quarterly – in electronic and printed form. All departments participate in the process of creating the quarterly. The most effective form of distribution

of magazines during the pandemic has become BISdelivery – sending out magazines ordered through a form on the website throughout Poland, in cooperation with the Post Office of AGH-UST. The Bulletin’s editorial staff also maintains a website that publishes both electronic editions of the magazines and self-contained articles.

An additional project carried out by BIS, in cooperation with the editorial office of Radio1.7, is the BIScast – a podcast in which audio versions of texts published in the magazines are created.

In order to identify the processes carried out by the members of the editorial team, the platforms for their mutual communication and the work tools they use were analysed. The most important functions they play in the work of the editorial office have also been identified.

A. Communication Platforms

1) Microsoft Teams

- organising online meetings,
- publishing meeting summaries.

2) Facebook

- publishing meeting summaries.

3) Facebook Messenger

- fast communication in private chats and group conversations.

B. Working Tools

1) Facebook

- publishing marketing material.

2) Instagram

- publishing marketing material.

3) Google Drive

- creating work schedules,
- creating publication schedules,
- storing files with article content and graphics.

4) Adobe Creative Cloud

- creating graphics,
- composing the bulletin and its parts.

5) Canva

- creating graphics for social media.

C. Quarterly Issue

Work on the quarterly begins with a meeting of the entire editorial team, at which the team members, first of all, agree on the theme and the leading color of the graphic design of the latest issue. At this meeting, the key deadlines for the submission of partial texts, their composition, and the deadline for the composition of the entire magazine are also established. After the meeting, all arrangements are written down in form of a post on the Microsoft Teams platform. The first modelled stage of the work is shown in Figure 2.

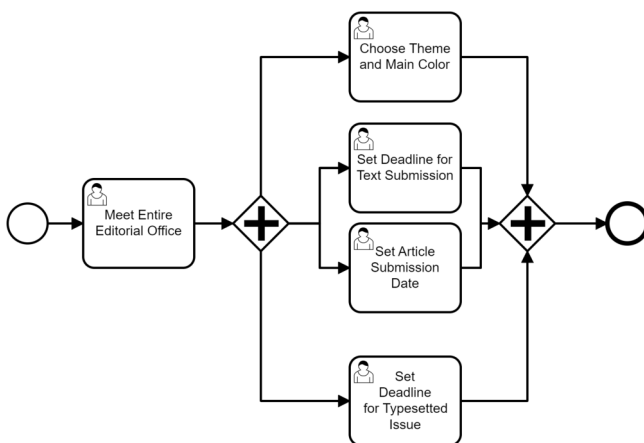


Figure 2. A BPMN model of the quarterly issue planning process

Next, the process of creating texts in the journalism section begins. The head of the department creates a table on Google Drive that will contain a list of texts for the latest issue. A meeting of the journalism department is held, where the table of topics is completed – each of the members and interns is to propose a topic for at least one article, summarize what the article is supposed to be about, specify the section to which it will be assigned, and complete their name in the "Who proposes?" column. The topics of the texts are free and they are divided into several categories. After the meeting, everyone has time to declare writing at least one of the proposed articles, and then the head of the department assigns proofreaders to specific texts.

The first step in the work of writing a text is to become familiar with its subject matter and necessary sources. Once the text is written, it should be uploaded to the appropriate folder on Google Drive. Later, one should also uncheck the

text upload in the table with texts. The corresponding record then turns red and the proofreader knows that the text is awaiting correction.

The text correction process then begins. Proofreading involves editing the document provided by the editor and giving the editor suggestions on how to avoid further errors, such as through the suggestion option in Google Documents. However, this option is imperfect in that the text creator must manually accept all corrections by the proofreader, or they will not be saved. The proofreader is also obliged to change the file name to "CORR_IN_title" (where IN are the first letters of the proofreader's first name and name) so that when one goes to the "Texts per page" folder, they will know for sure which texts have been corrected. After language proofreading, the proofreader should also uncheck the checkbox in the "Proofreading" column.

After defining all the activities and their sequence, the identification of sub-processes has been conducted. For this task, we used the method called Justified Aggregation of Neighboring Activities, the idea of which is based on graph models [24]. In order to declare a set of activities as a sub-process, the following conditions must be met:

- 1) Activities are connected with a sequence flow.
- 2) There is one clear starting point of the candidate sub-process.
- 3) Outside boundary activities are defined. In order to be considered an outside boundary activity, task or sub-process has to fulfill at least one of the three conditions:
 - its meaning is not related to any of the tasks already included in the candidate sub-process,
 - its responsible function (represented by annotation or a swimlane) is not present in the candidate sub-process,
 - in case of multi instance or recurring activities, its instances do not match with the other activities in the candidate sub-process

The modelled process can be found in Figure 3. One of the identified sub-processes is "Write Text". In this case, *Assign Proofreaders* has been defined as an outside boundary activity, as it is executed by another process stakeholder and is not a multi-instance activity, unlike the other tasks included in the sub-process. The second outside boundary activity is *Proofread Text*. Although it is also a multi-instance activity, its instances involve different participants than those in the case of proofreading.

When all the texts for the quarterly journal are ready, a table is created by the head of the graphics department to divide the further work. The titles and sections of the texts are copied from the table of the journalistic department. For convenience, links to specific documents from the journal section folder are also pasted. The chief then distributes the illustration and composition of the texts among the members and staff.

The issue's editorial cycle includes cover design, photo pages, and cover art for later published podcasts. The cover

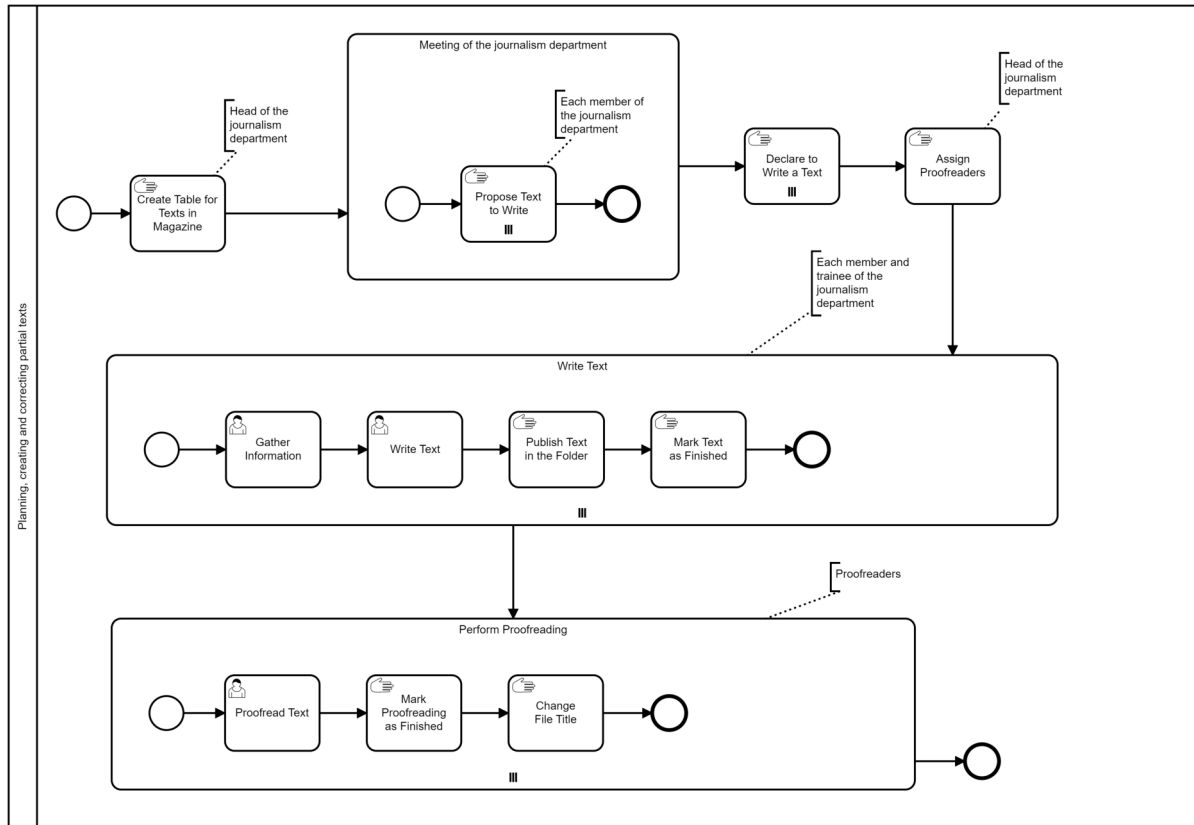


Figure 3. The process model of planning, creating and correcting partial texts

photo and photos for the photo pages are obtained by the editors from the Krakow Student Photo Agency. Usually, the photos are also placed on Disk Google.

By the previously agreed date, illustrations for the text and other materials, exported separately in Adobe Illustrator and text filed in Adobe InDesign, should appear in the Google Drive folder. No additional form of proof of submission materials is required.

The next step is the composition of the entire magazine, which the Editor-in-Chief usually does. He then processes the files obtained from all the graphic designers and creates one, preparing the magazine for printing. The modelled process can be seen in Figure 4.

When the quarterly magazine is ready for printing, the preparation of a subpage with its content on the website follows and the preparation of promotional materials begins. In order to plan the work, the promotion and collaboration department uses a spreadsheet, designed in the shape of a calendar. The calendar schedules posts for the following weeks. Often, however, the information in the spreadsheet is inaccurate or selective. First, the type of material to be published regarding the text is determined - a photo or a graphic. This is followed by scheduling the publication of material for specific days and assigning people to perform specific tasks. Materials for posts are prepared in Canva, where

the promotion department works as a team. This means that members can share projects with each other, create templates or use a common content planner. There, posts are created in their entirety because the service also gives the ability to add images and descriptions to specific posts. The final step in the development of the quarterly journal is its distribution. The first form of distribution was direct delivery of printed copies to university departments, student house receptions, and dining facilities. Currently, the distribution consists mainly in sending the magazines (in cooperation with the AGH-UST Post Office) throughout Poland. Orders are collected using a form on the internet website.

D. Website

Only those in charge have the ability to edit the editorial page. Three times a week (on Monday, Wednesday, and Friday), new articles are posted on the website and - once every three months - a subpage with the new quarterly issue.

Each member and intern of the journalism department is required to write at least two articles for the website per month. Once a month, during a meeting, everyone completes their proposed topics in the appropriate table on Google Drive. Afterwards, the editors have time to commit to writing an article (either their own or someone else's). Then the head of the department establishes the schedule for the publication of the texts on the site, and according to this criterion, determines

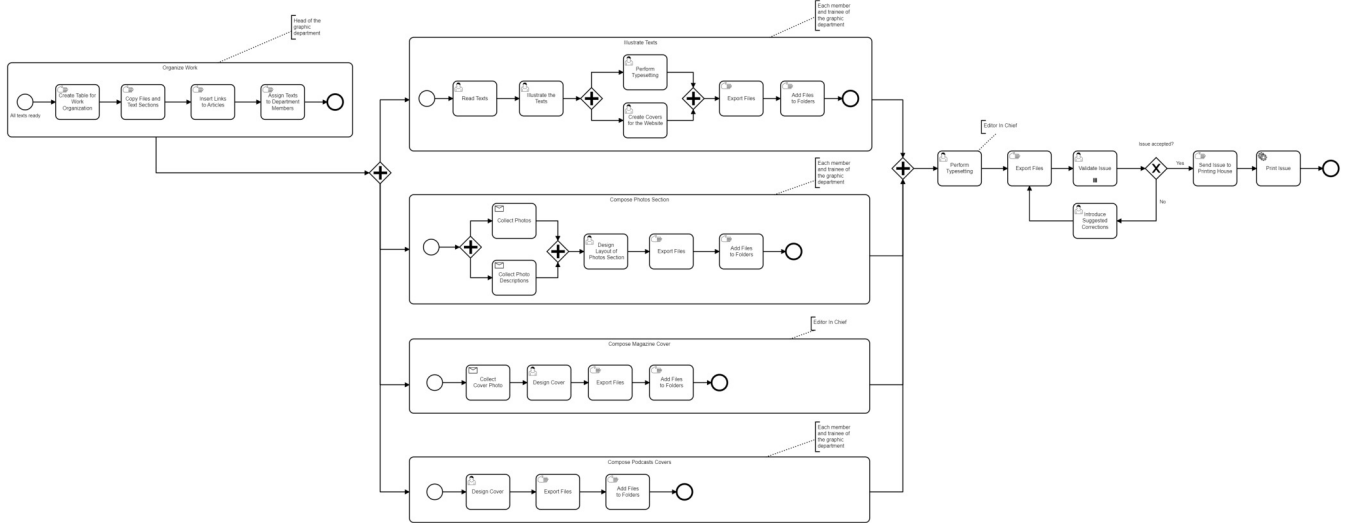


Figure 4. The process of planning, illustrating, and composing the issue and its parts

when the text should appear on the shared drive, and who is responsible for its correction and publication.

Proofreading is a very important part of the process, without which the text cannot be published on the site. Language proofreading is done by three designated individuals in the department. When scheduling articles for publication, they are also assigned to do proofreading of specific articles.

The article-writing process then begins, which is always preceded by gathering information on the topic. After submitting an article to the disk (as in the table on texts for the quarterly journal), editors check the checkbox located to the left of the record with their text. The record then turns blue and the person assigned to language proofreading knows that the text is awaiting correction. After the linguistic correction, the proofreader should also uncheck the checkbox in the "Correction" column so that the person publishing the article knows that it is ready for publication (after which they should also check the appropriate checkbox indicating that the text has been published).

It turns out that a characteristic system error is that editors often forget to check boxes after the task is done, making the system ineffective because it does not provide reliable information. Therefore, all information must be independently verified, both before proofreading texts and publishing them. The modelled process can be seen in Figure 5.

The graphics department is not involved in the creation of posts for the website; once published, the link to the text is usually shared on Facebook and in Instagrams stories, which are handled by members and interns of the promotions and collaboration department. The posts usually appear on the same days as the texts, namely on Mondays, Wednesdays, and Fridays. The head of the promotion department distributes the posts to the editors on a regular basis, usually by sending them a private message on Facebook with a request to publish on a given day.

The promotions department uses mostly the same spreadsheet to organise their work as they do to create the promotion strategy for the quarterly magazine. The spreadsheet, however, shows only perfunctory information regarding what type of material is published on what day. It lacks information about who is responsible for publishing what materials and when. The table is also not updated on an ongoing basis, so many materials in it are not there and are published spontaneously.

V. DESIGN OF THE SYSTEM INTERFACE

After creating the information architecture diagrams of the panels of each role, their mock-ups and prototype of the system with a medium level of detail were created. This chapter presents a schematic design of the screens that are key to carrying out the processes performed in the editorial office and departments. The layout of each page, the placement of action buttons, labels, and other elements are shown. The prototype was made in grayscale with color coding in places where the type of color indicates the status of the action.

A. Main Page and Task List

Figure 6 shows the screen that is visible after logging in to the panel for users with the role, "Journalism Department". At the very top of the side navigation bar, located on the left side of the screen, are the user's information - first name, last name, department (this information is completed by the administrator when creating a new account), and profile picture (which can be added and changed by the user by clicking on their avatar).

B. Quarterly Issue

Once the theme and color of the quarterly layout have been completed by the administrator, the screen shown in Figure 7 appears in the panel of the journalism department, under the tab, "Quarterly".

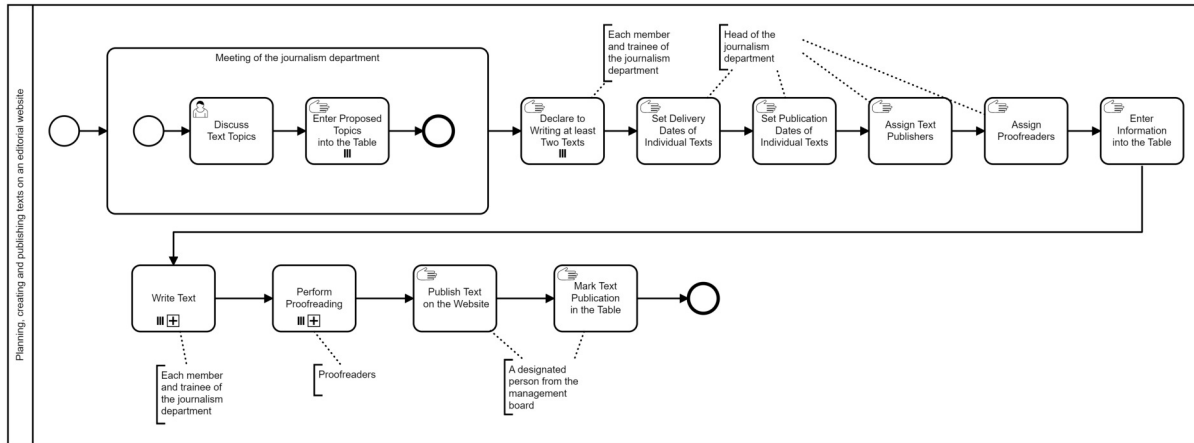


Figure 5. The process model of planning, creating and publishing texts on an editorial website

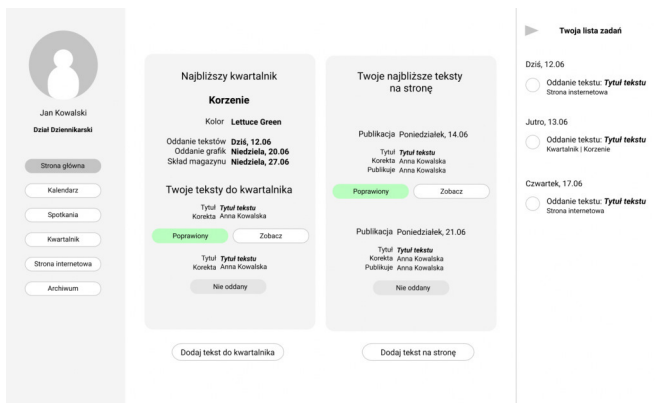


Figure 6. The view of the „Strona główna” (“Home”) tab in the user panel with the role „Dział Dziennikarski” (“Journalism Department”)

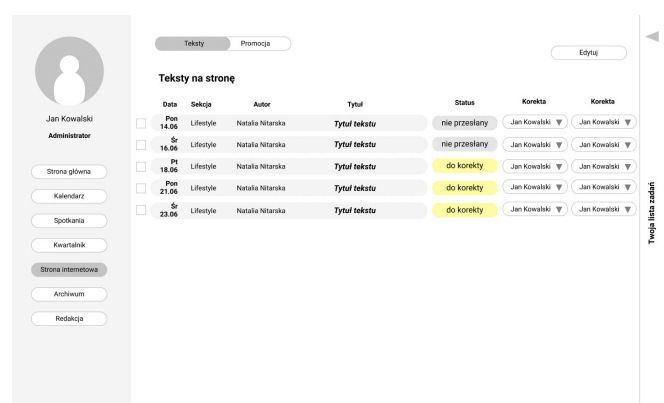


Figure 8. The view of the „Teksty” (“Texts”) subpage in the „Strona internetowa” (“Website”) tab in the user panel of the role „Administrator”

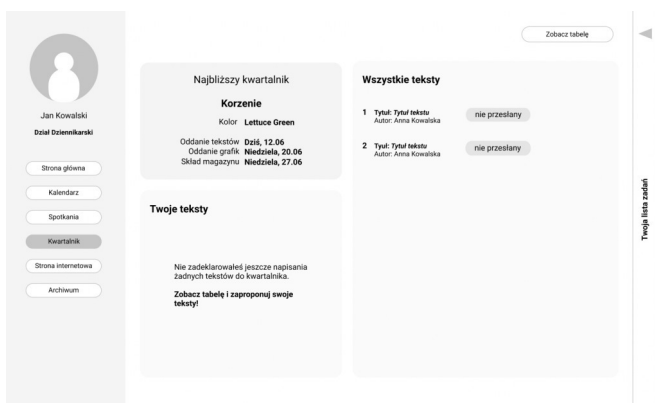


Figure 7. The view of the „Kwartalnik” (“Quarterly”) tab in the user panel of the „Dział Dziennikarski” (“Journalism Department”) role after adding a new quarterly

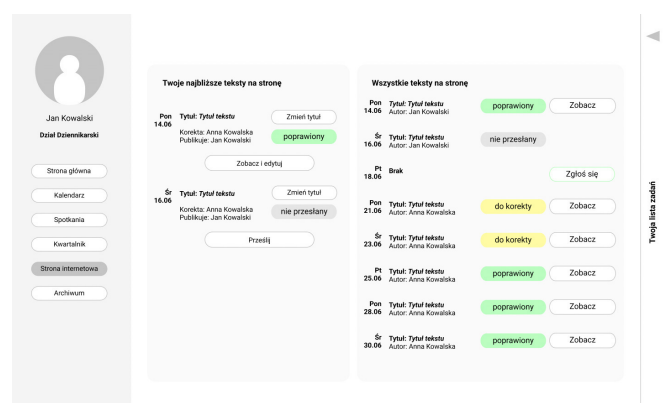


Figure 9. The view of the „Strona internetowa” (“Website”) tab in the user panel of a role „Dział Dziennikarski” (“Journalism Department”)

In the upper left corner of the main part of the screen is a banner with basic information about the upcoming quarterly. Underneath it is a section that displays the texts to which the

user has assigned himself as author. On the right side of the screen there is a section called "All texts", in which all texts for the upcoming quarterly to which an author has contributed are displayed.

C. Website

Figure 8 shows the „Website” tab of the admin panel.

The administrator, using the "Edit" button in the upper right corner of the screen can not only edit record data, but also add new ones according to the website's text publication schedule. By default, a new record will appear for every Monday, Wednesday or Friday, but this can be freely modified by the administrator.

Figure 9 shows the "Website" tab in the panel of the journalism department. The tab is divided into two parts.

On the left side of the screen, there is information about the nearest texts for the website to which the user is assigned – planned publication date, title, who is publishing, and status. Depending on the status of the assignment, there are also buttons for , "Submit" or "View and edit".

On the right side of the screen, information about all planned texts for the website is available, sorted by publication date (from nearest to farthest).

If there are no editors assigned to a particular date, there is a button next to it called "Submit", which the user can use to declare writing a text for a particular date.

VI. CONCLUSIONS

The purpose of this paper was to present step by step our case study of designing a system user interface that greatly enhances and organises the flow of information among members and collaborators of the editorial board of the AGH Student News Bulletin. With the use of various research methods such as observation, social research, and process exploration methods, we analysed the activities executed by the editorial board, communication platforms, and members' working tools. Using knowledge of user interface design, we designed a system interface that effectively reduces the need to use many of these tools and provides a platform through which most of the information needed by the editorial board can be collected and transmitted.

In order to effectively accomplish the stated goal and properly design the system interface, we focused on one of the four editorial offices. This enabled accurate data analysis and the elimination of errors. Using the lessons learned during the research phase, it was possible to design an interface to support the work of the entire organisation.

Further possible activities in the development of the project might be focused on conducting usability tests of the prototype with members of the Bulletin editorial staff. After testing the designed solutions and making necessary corrections, it would be necessary to develop the graphic layer of the project and develop both the front-end (based on the designed user interface) and the back-end for the presented system.

REFERENCES

- [1] D. Knemeyer and E. Svoboda, "User Experience-UX in The Glossary of Human Computer Interaction," *The Interaction Design Foundation*, 2015.
- [2] S. R. Stein, "The "1984" macintosh ad: Cinematic icons and constitutive rhetoric in the launch of a new machine," *Quarterly Journal of Speech*, vol. 88, no. 2, pp. 169–192, 2002.
- [3] A. Dix, A. Dix, J. Finlay, G. Abowd, and R. Beale, *Human-computer Interaction*. Pearson/Prentice-Hall, 2003.
- [4] B. Shneiderman, "Designing for fun: how can we design user interfaces to be more fun?" *interactions*, vol. 11, no. 5, pp. 48–50, 2004.
- [5] —, *Designing The User Interface: Strategies for Effective Human-Computer Interaction, 4/e (New Edition)*. Pearson Education, 1987.
- [6] E. Wong, "Shneiderman's eight golden rules will help you design better interfaces," <https://www.interaction-design.org/literature/article/shneiderman-eight-golden-rules-will-help-you-design-better-interfaces>, dostę: 2021-04-21.
- [7] J. Nielsen, "Ten usability heuristics," <https://www.nngroup.com/articles/ten-usability-heuristics/>, dostę: 2021-04-21.
- [8] N. N. Group, "Nn/g history," <https://www.nngroup.com/about/history/>, dostę: 2021-07-03.
- [9] D. A. Norman, *The Design of Everyday Things*. Currency Doubleday, New York, 2013.
- [10] S. Harwood and N. Hafezieh, "'affordance'-what does this mean?" in *22nd UK Academy for Information Systems International Conference: Ubiquitous Information Systems: Surviving & Thriving in a Connected Society Oxford*. St. Catherine's College Oxford, UK, 2017.
- [11] J. Nielsen, "Mental models," <https://www.nngroup.com/articles/mental-models/>, dostę: 2021-04-21.
- [12] J.-R. Rehse and P. Fettke, "Clustering business process activities for identifying reference model components," in *International Conference on Business Process Management*. Springer, 2018, pp. 5–17.
- [13] I. Mościchowska and B. Rogoś-Turek, *Badania jako podstawa projektowania User Experience*. Wydawnictwo Naukowe PWN SA, 2015.
- [14] A. Moghaddam and G. Gadanidis, "Designing an online collaboration system," in *EdMedia+ Innovate Learning*. Association for the Advancement of Computing in Education (AACE), 2005, pp. 548–553.
- [15] K. Marasek, A. Romanowski, and M. Sikorski, "Emerging trends and novel approaches in interaction design," in *2017 Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2017, pp. 1231–1234.
- [16] E. Franchi, A. Poggi, and M. Tomaiuolo, "Social media for online collaboration in firms and organizations," in *Information Diffusion Management and Knowledge Sharing: Breakthroughs in Research and Practice*. IGI Global, 2020, pp. 473–489.
- [17] J. Widén and M. Johansson, "BPMN flows as variation points for end user development: from a ux perspective," 2016.
- [18] E. Diaz, J. I. Panach, S. Rueda, and O. Pastor, "Towards a method to generate GUI prototypes from BPMN," in *2018 12th International Conference on Research Challenges in Information Science (RCIS)*. IEEE, 2018, pp. 1–12.
- [19] E. Diaz and S. Rueda, "Generation of user interfaces from business process model notation (BPMN)," in *Proceedings of the ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, 2019, pp. 1–5.
- [20] C. dos Santos Garcia, A. Meinheim, E. R. F. Junior, M. R. Dallagassa, D. M. V. Sato, D. R. Carvalho, E. A. P. Santos, and E. E. Scalabrin, "Process mining techniques and applications—a systematic mapping study," *Expert Systems with Applications*, vol. 133, pp. 260–295, 2019.
- [21] P. Weichbroth, M. Owoc, and M. Pleszkun, "Web user navigation patterns discovery from www server log files," in *2012 Federated Conference on Computer Science and Information Systems (FedCSIS)*. IEEE, 2012, pp. 1171–1176.
- [22] L. S. Rosa, T. S. Silva, M. Fantinato, and L. H. Thom, "A visual approach for identification and annotation of business process elements in process descriptions," *Computer Standards & Interfaces*, vol. 81, p. 103601, 2022.
- [23] M. Nizioł, P. Wisniewski, K. Kluza, and A. Ligeza, "Characteristic and comparison of UML, BPMN and EPC based on process models of a training company," *Annals of Computer Science and Information Systems*, vol. 26, pp. 193–200, 2021.
- [24] D. Zhang, L. Liu, Q. Wei, Y. Yang, P. Yang, and Q. Liu, "Neighborhood aggregation collaborative filtering based on knowledge graph," *Applied Sciences*, vol. 10, no. 11, p. 3818, 2020.

The impact of the multi-variant remote work model on knowledge management in enterprises. Applied tools.

mgr inż. Anna Nowacka
Faculty of Management,
Czestochowa University of Technology,
9B Armii Krajowej Street,
42-200 Czestochowa, Poland
Email: anna.nowacka@pcz.pl

prof. dr hab. Dorota Jelonek
Faculty of Management,
Czestochowa University of Technology,
19B Armii Krajowej Street,
42-200 Czestochowa, Poland
Email: dorota.jelonek@pcz.pl

Abstract—The article presents the scope of issues related to knowledge management in models assuming the performance of work outside workplaces and the tools used to disseminate knowledge transfer. Knowledge management and transfer are the cornerstones of any business. They are evidence of market position and competitive advantage. The approach to knowledge transfer tools in remote work models is becoming crucial. On this basis, a research issue was formulated, which concerns knowledge management in hybrid work performance models and ICT tools used. The research objective is to assess knowledge transfer in businesses and identify the most effective ICT tools used by employees to manage knowledge in businesses. In order to solve the research problem, quantitative research was conducted on a randomly selected sample of respondents. Studies have shown that respondents are aware of the importance of knowledge sharing within organizations. They see the benefits and barriers thereof. Their command of knowledge-sharing tools, on the other hand, is moderate. The added value of the article is linking the substantive approach to knowledge transfer itself, taking into account the tools used in the hybrid model. The article indicates the existing connections and perception of the employees themselves.

Index Terms—knowledge, knowledge management, hybrid work model, ICT tools.

I. INTRODUCTION

MODELS of performing work outside the employer's headquarters have gained importance due to the global coronavirus pandemic. This enabled the emergence of a new reality that encompassed all possible employers. This determined the definition - in accordance with the SWOT analysis - of new opportunities, threats, as well as specifying the strengths and weaknesses of this model. Many companies do not have clearly defined strategies for applying remote working models. This raises many doubts among employees, since the coronavirus pandemic has shown that many tasks and duties can be performed in remote models. This is also directly related to maintaining work-life balance, which is becoming increasingly important. There is a growing attention being paid to creating a culture of knowledge sharing by employees in businesses. A culture that is based on openness, trust and a favourable atmosphere. Knowledge is the primary resource of an organization, which testifies to

its market position and competitiveness. Resources, on the other hand, are employees, and in particular their skills and competences, which have open or secret forms of knowledge. The objective of businesses is to create such conditions in which employees will want to exchange their experience and know-how. To make this possible, it is necessary to use tools that will make it possible. These are primarily systems or knowledge bases that employees have access to. The article aims to analyze the knowledge transfer tools used in hybrid work models in businesses.

The article reviews the current literature on the subject, presents the research process methodology and the characteristics of the research sample, presents the research results, refers to the latest report results, and finally summarizes the conclusions.

The author's contribution to the article is as follows:

- carrying out a critical analysis of the research topic,
- referencing the subject matter of the article in the literature overview,
- conducting quantitative research on a random sample of respondents,
- research results constitute added value in the field of management and quality sciences. They indicate the connections in the examined topic.

II. LITERATURE OVERVIEW

The concept of remote work or telework/telecommuting dates back to the 1970s. This was a response to many changes in the processes within businesses. This resulted in the emergence of innovative approaches to shaping new models of work performed outside the employer's headquarters. In 1973, Jack Nilles conducted an experiment that proved that each employee's work can be transmitted through computers or other telecommunications methods. This, in consequence, meant limiting the movement of workers who are required to perform work [1]. This was greatly aided by technological development, which determined the emergence of a new and flexible employment form for employees who could carry out their tasks outside the employer's stationary offices. The approach aims to limit com-

muting costs in addition to saving on commuting time [2]. The possibility of performing duties outside the workplace has been intensified by the increased importance of information and knowledge in relation to goods or products. The development of remote work is the development of information technologies as well as computer science itself, which is a tool in IT architecture [3].

There are many definitions of the term 'remote work'. It is very difficult to cite one that would fully reflect the essence of its meaning. In order to attempt to define remote work, one should draw attention to existing synonyms such as teleworking, virtual officing, remote work, networking, or work from home. It seems that the terms can be used interchangeably, because they mean the same thing. The difference may exist in the basis of employment under which work is performed [4].

TABLE I
Selected definitions of the term of teleworking

A. Jeran (2016)	"Work carried out outside the employer's offices, depending on the form: in the employee's place of residence or another location, sometimes on the move" [5].
M. Hynes (2014)	The possibility of using IT tools that allow one to perform their duties in a different geographical location than the workplace [6].
R. Blanpain (2013)	"Work performed for an employer or client, mainly in a place other than the traditional workplace, using information technologies" [7].
Greenberg and A. Nilssen (2008)	Implementation of new information technologies through tools, including computers, in exchange for employees commuting to the workplace [8].
S. Ciupa (2009)	"A new form of organizing and performing work (...), where the manner and conditions of performance as well as order and organization can be shaped through the use of advanced information and communication technologies" [9].
The Act of 26 June 1974, art. 67, § 1 and 2. Labour Code (1974)	"It may be performed regularly outside the workplace, using electronic means of communication in light of regulations on the rendering of electronic services (teleworking). The teleworker is an employee who performs work (...) and provides the employer with the results of said work, in particular using electronic means of communication" [10].

Source: Own study based on the cited literature

Remote work is a type of flexible work in the scope of dimensions such as: time, location, permanence of relation-

ships, and the type of contracts concluded between the employer and the employee [11].

According to S. Hogarty, the hybrid work model aims to combine elements characteristic of office-based work and performing duties as part of remote work. The goal is to provide the employee with an opportunity to decide how they will perform their work. The hybrid system of work is based on the performance of tasks by employees depending on their preferences and work style. Each employee must have full comfort in performing their duties. There are the following types of hybrid work: "remote first", which means that employees perform their tasks remotely, but can come to the office if necessary. The rotation variant, which consists in establishing a schedule and dividing employees into teams performing work remotely and stationary [12].

The advantages of using a hybrid work model include [13]:

- the possibility of saving time on commuting to the workplace,
- growing importance of work-life balance for employees,
- increase in KPIs in terms of employee satisfaction with and engagement in the performed work,
- greater flexibility in the recruitment carried out by companies, as well as reaching out to "talent",
- ability of maintaining better contacts with clients.

Knowledge management is becoming increasingly important in modern businesses as a foundation of the human resources management strategy. As a result of the end of the industrial era, some of the most important assets are knowledge, information and intellectual capital [14]. The issue at stake is the knowledge of employees itself, which is not the property of the companies [15]. Intensified development of the information society focusing on the knowledge management and distribution plays a key role in all communication progress. Telecommunications and all information technologies determine contacts between people and have a direct impact on teamwork in distributed structures [16].

The word "knowledge" has many definitions in subject literature. According to P. Ducker - knowledge is a type of effective and efficient use of information in taking actions [17]. On the other hand, E. Turban believes that components of knowledge include: truth, beliefs, expectations, ideas, and know-how [18]. S. Galata argues that knowledge is an exceptional and unique resource of an organization that accrues while in use [19].

One of the trends of innovative employee behaviors is the use of creativity based on three competencies, which include: expertise, motivation and creative abilities [20].

Human and intellectual resources have a key impact on building the innovation capacity of businesses where knowledge is a key element. Knowledge management, then, is influenced by the following:

- the education of employees hired by the business,
- skills related to knowledge acquisition and processing,
- openness of employees to improve their qualifications, as well as acquire new knowledge,

- acquired knowledge and experience of the management staff,
- all knowledge regarding the needs of internal and external clients, as well as contractors,
- competency opportunities to create new knowledge,
- cumulative knowledge (codified knowledge) encompassing intellectual property rights,
- possessing competencies related to knowledge acquisition from the external environment [21].

A given organization's management staff will support the knowledge management process, including the knowledge transfer itself, if it is related to obtaining potential benefits [22].

Knowledge management in the Japanese model consists of a process that includes the following elements:

- socialization – assumes experience sharing between employees. Creating hidden knowledge, as well as generating new ideas.
- externalization – that is, manifestation. It is a situation in which hidden knowledge is presented as a metaphor, a hypothesis. Afterwards, it is converted into overt knowledge.
- combination – combining available knowledge through various components. The result is the reaction of new knowledge.
- internalization – involves the permeation of available knowledge into hidden knowledge. It primarily involves education through taking action [23].

The model presented above is the beginning of the SCRUM concepts, which concern the concept of managing organizations in the perspective of project management. It is an operational level based on planning, as opposed to a traditional concept based on control [24].

Hidden knowledge is a type of knowledge that is difficult to articulate through colloquial language [25]. Overt knowledge is created on the foundations of hidden knowledge, which serves as the starting point [26].

In the process model, knowledge management integrates processes such as creating, organizing, collecting, and then utilizing knowledge in the organization [27].

If knowledge management in the process approach constitutes a process, it is possible to distinguish the following sub-processes:

- creation of new knowledge (recognition of new and old knowledge),
- specification of knowledge belonging to the organization,
- checking the knowledge resources,
- ordination of knowledge,
- the possibility of utilizing and promoting knowledge,
- creation of knowledge based on resources [28].

The resource model is based primarily on resources as the basic factor indicating a competitive advantage [29]. However resources, including knowledge, are combined when creating the competitive advantage itself. This determines an exchange of knowledge between employees [30].

Knowledge transfer support tools within organizations include:

- a document circulation system (workflow) that allows for circulation of documents from different sources, but existing within one system, which all employees involved in a given process have access to. The scope of tasks includes: managing document flow, i.e. document life, data import / export.
- a system aimed at supporting group work, which enables joint performance of tasks or projects in distributed structures or in hybrid work models.
- an expert system that enables the use of available knowledge, and related decision-making. The scope of this system includes: a user interface, a knowledge base, a database, and a knowledge base editor.
- an e-learning system for monitoring and reporting progress results in the remote learning of employees.
- a content management system is a tool intended to create websites in the HTML model [31].

The figure below shows the relationship between knowledge management, which, using appropriate tools, affects the work performance in a hybrid model.

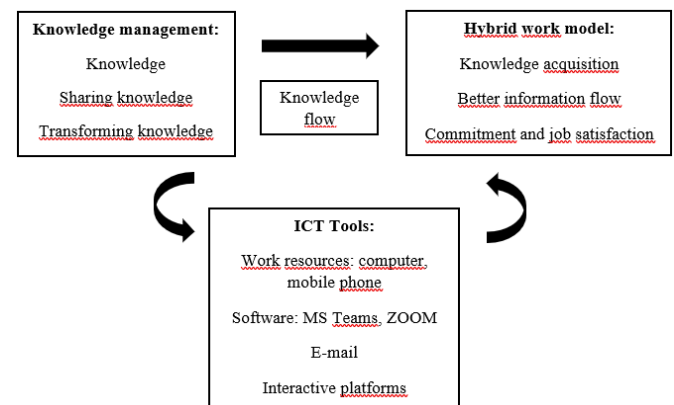


Fig. 1 Relationships between knowledge management and the hybrid work model
Source: Own study

For the purposes of the article, a multi-variant remote work model has been developed, which has a direct impact on knowledge management in businesses. The model was created on the following models: resource-based, process-based and Japanese.

The model's assumptions are as follows:

- knowledge is the strategic and most important pillar of a business,
- knowledge exchange takes place among employees working in a hybrid model,
- knowledge has a multifaceted dimension,
- managing knowledge is a process consisting of its creation, transfer, implementation, sharing, and transformation.
- knowledge is not only information, but also values, emotions, views, and experience,
- ICT tools absolutely affect the transfer of knowledge between employees.

III. METHODS AND MATERIALS

Based on the theoretical section, the following objectives of the conducted research were formulated:

- assessment of tools used by employees during hybrid work,
- assessment of the hybrid work model in businesses,
- assessment of knowledge transfer in businesses.

The article puts forward the following research hypotheses:

H1: Knowledge sharing has a significant impact on employees of companies.

H2: The use of ICT tools positively impacts the use of knowledge in businesses.

H3: Knowledge management depends on the multi-variant model of remote work adopted in the article.

In order to achieve the research objectives, a critical analysis of the subject literature was conducted. Literature studies were conducted using the desk research method utilizing secondary sources. Their objective was to prepare a research tool for conducting empirical analyses. The research was quantitative in nature. A prepared online survey was used as the research tool. The survey was conducted on 08-31.03.2022. Selection of the research sample was random. The study was anonymous and participation in it was voluntary.

The questionnaire consisted of 20 substantive questions and a metric. Some of the questions were open, asking the employee to provide an individual answer, while others were closed questions with the option of choosing one or several answers. The survey was taken by 239 respondents.

TABLE II
Survey results – metric

No	Category	Respondents' answers	Answers In figures	Answers in [%]
1	Gender	Female	150	62,8%
		Male	89	37,2%
2	Age	Under 25 years old	42	17,6%
		25 - 40	90	37,7%
		41 - 60	64	26,8%
		Over 60	43	18%
3	Education	Primary	27	11,3%
		Secondary	133	55,6%
		Higher	79	33,1%
4	Company size	Small	124	51,9%
		Medium	59	24,7%
		Large	56	23,4%

Source: Own study

It should be noted that the research was carried out after a subsequent wave of the COVID-19 pandemic, which lasted from January to February 2022. There is a return of employees to stationary or hybrid work observed in businesses.

IV. RESULTS OF THE CONDUCTED STUDY

The growing popularity of hybrid work requires companies to provide employees with the tools to perform their jobs outside the employer's offices.

The answers to open question concerning the tools necessary to perform duties in a hybrid model are presented in Figure 2. Respondents stated that the computer is a priority (38.3%). It's a basic tool that directly enables performance of work regardless of the employees' geographic location. The knowledge of internet tools enabling remote connection of employees – MS Teams, ZOOM (24.8%) was highly rated. Internet access (19.1%), as well as a mobile phone (5.3%), headphones with a microphone (4.4%), or other electronic devices (0.4%) are also considered necessary. A small number of respondents answered that this question did not apply to them (2.7%).

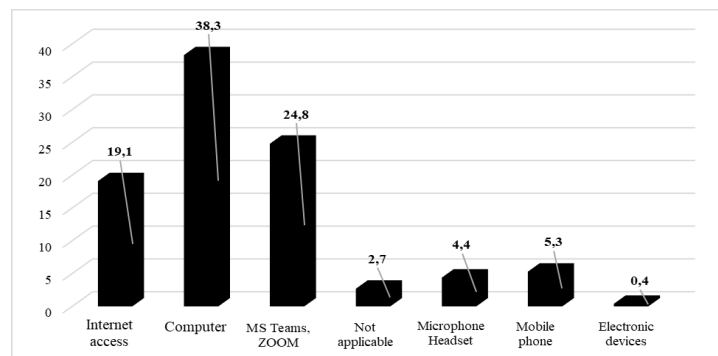


Fig. 2 Tools necessary to perform duties in a hybrid work model
Source: Own study

Figure 3 presents the factors that respondents believe are in favor of performing work in a stationary model. Over half of the respondents consider contact with other colleagues to be the most important (52.8%). For employees, another important aspect is the greater control they have at work (12.4%) and the ability to focus on their job (6.8%). An equally important issue is access to all tools (8%) such as a printer, scanner or documents, which are only available in the workplace. Respondents also drew attention to the workplace atmosphere (7.2%). This is an extremely important factor which shows that employees feel the need to be at work in a pleasant atmosphere. There were also answers concerning: greater discipline (4%) and willingness to help others (3.2%).

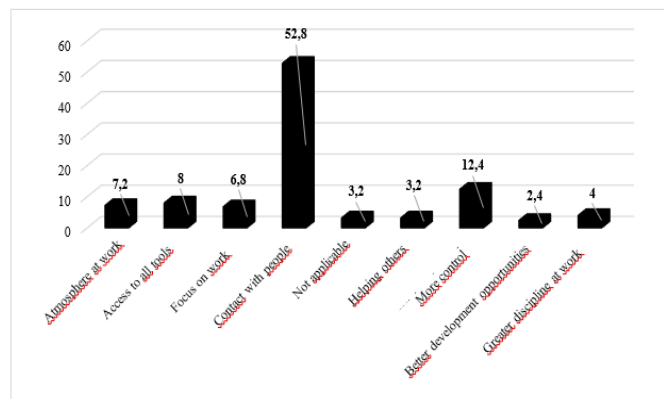


Fig. 3 Factors affecting stationary office work
Source: Own study

The next question in the survey questionnaire concerned the ranking factors that affect job satisfaction in the hybrid model from the most important to the least important. According to respondents, the most important is flexibility, which is closely related to performing tasks outside the employer's headquarters (48.1%). In line with the theory by Jack Nills, respondents note a reduction in time associated with commuting (38.5%). When working from home, an important factor turns out to be the employer's trust in the employee reliably and conscientiously performing their work (29.7%). The answer to this question is related to the following answer of respondents concerning the maintenance of work-life balance (14.6%). Limiting the commute time, along with the employer's full trust in the employee performing their tasks directly affects the ability to maintain work-life balance. For workplaces with an open space, it is important that working from home allows for greater focus (19.7%).

Table 3 presents the results of respondents' answers to questions concerning work models implemented in businesses. The distribution of answers to the question related to the work mode's efficiency is very similar in terms of the stationary work model (36.8) and the hybrid work model (36%). The answers differ in the case of the work model applied in businesses. According to the respondents' answers, organizations prefer to work in a stationary model (44.8%) assuming that employees come to the employer's headquarters to perform their duties. The research shows that hybrid and remote work models are not popular among employers of the randomly surveyed respondent sample. Over half of the respondents (67.8%) believe that meetings with colleagues are a key element of teamwork. Respondents believe that the companies where are employed have provided them with appropriate knowledge and ensured corresponding knowledge carriers to perform their duties.

TABLE III
Results of the survey concerning the performed work model

No.	Question	Answers to the questions asked in the survey	Respondents' answers	Average value	Median	Standard Deviation
1	What mode of work do you consider to be the most effective?	Stationary work model	88	79,67	86	10,40
		Hybrid work model (combining remote and stationary work)	86			
		Remote work model	65			
2	What is the work model solution in your organization?	Stationary work model	107	79,67	91	28,11
		Hybrid work model (combining remote and stationary work)	91			
		Remote work model	41			
3	Do you believe it is necessary to meet your team live?	Yes	162	119,5	162	42,5
		No	77			
4	When performing professional duties from home, do you think that the company has appropriately adapted the transfer of knowledge?	Yes	146	119,5	146	26,5
		No	93			

Source: Own study

Table 4 presents the respondents' answers to questions concerning knowledge transfer within employee relations in businesses. A very important element here seems to be the exchange of knowledge between employees, because it is through their work and effort that organizations implement business strategies and goals. The know-how of employees forms the basis for maintaining business continuity as well as company secrecy. The results presented below show that knowledge transfers are moderate between all employees at different levels in organizations. Most respondents (38.5%) believe that the best knowledge flow occurs in the manager-employee relationship. Top-down communication in this configuration testifies to the transfer of the supervisor's knowledge to their subordinates, including on what task to perform, what data is required, and often these are conversations regarding organizational changes. Communication in the employee-employee relationship was also assessed by respondents as moderate – 33.9%. Respondents gave the worst rating to bottom-up communication, i.e. the flow of knowledge between the employee and the manager – 31.8%.

Knowledge transfer in hybrid work models is primarily the knowledge of digital tools enabling communication between colleagues and the transmission of work results. It is a basic skill determining the digital competencies that are gaining so much importance. The majority of respondents assess their knowledge of online tools at a moderate level (30.5%). Only 11.3% assess their knowledge as very good, while as many as 16.7% of respondents practically do not deal with these types of tools.

TABLE IV
Results of the survey concerning knowledge transfer

No.	Question	Scale (1 – very weak, 2 – weak, 3 – moderate, 4 – good, 5 – very good)				
		1	2	3	4	5
1	How do you assess the flow of knowledge in the employee-to-employee relationship in the hybrid model	38 15,9%	41 17,2%	81 33,9%	54 22,6%	25 10,5%
2	How do you assess the flow of knowledge in the manager-employee relationship in the hybrid model	30 12,6%	36 15,1%	92 38,5%	52 21,8%	29 12,1%
3	How do you assess the flow of knowledge in the employee-manager relationship in the hybrid model	32 13,4%	37 15,5%	76 31,8%	68 28,5%	26 10,9%
4	What is your knowledge of digital tools, e.g. digital platforms, including MS Teams, ZOOM, ClickMeeting	40 16,7%	45 18,8%	73 30,5%	54 22,6%	27 11,3%

Source: Own study

Table 5 presents the correlation results of questions regarding knowledge transfer according to Spearman's ranks. The correlation between the variables is at a very low level.

TABLE V
Spearman's index for the survey results concerning
knowledge transfer

No.	Question	Spearman's rank correlation coefficient - r,
1	How do you assess the flow of knowledge in the employee-to-employee relationship in the hybrid model	-0,1
2	How do you assess the flow of knowledge in the manager-employee relationship in the hybrid model	-0,1
3	How do you assess the flow of knowledge in the employee-manager relationship in the hybrid model	-0,1
4	What is your knowledge of digital tools, e.g. digital platforms, including MS Teams, ZOOM, ClickMeeting	-0,1

Source: Own study

Table 6 presents the results of respondents' answers to questions concerning knowledge management in businesses. The following questions are multiple choice. The respondents believe that the most important sources of knowledge transfer are external training courses (47.7%) and knowledge exchange between employees (44.8%). Respondents use resources such as knowledge repositories in organizations (10.9%) or market research reports (16.3%) the least. As the most important factors influencing knowledge sharing within the organization one may consider: employee trust (46.9%) and respect towards employees (46.9%). It follows that for employees the most as important fundamental the values that directly affect their openness in sharing knowledge. Remuneration was also assessed very highly (39.7%) as a financial motivator. It can be hypothetically assumed that the higher the employees' remuneration, the greater their willingness to share knowledge. The company's organizational culture was rated the lowest – 16.3%. In the subsequent question, respondents had to choose statements that describe their enterprises to the greatest extent, and above all, the organization's attitude to knowledge management. The largest number of respondents (34.7%) replied that knowledge transfer takes place mainly between employees and the knowledge sharing and management within organizations is a key process of the functioning of companies on the market (32.2%). Employees are the greatest source of knowledge in companies, and knowledge exchange is a fundamental element of their business. It should be noted that despite the awareness that the knowledge transfer-related processes are very important and both the management and employees know about it perfectly, simply promotion of a knowledge sharing culture is imperceptible which is evidenced by the respondents' answers in this regard – 12.1%.

This implies that organizations devote insufficient efforts to further intensify the dissemination of the knowledge sharing culture at all organizational levels. In terms of tools used to disseminate knowledge in enterprises, these are: interactive platforms (including e-learning platforms) – 30.1%, e-mails (used by employees to send each other their know-how) – 31%, internal employer tools (which, among others, include

TABLE VI
Results of the survey concerning knowledge management
knowledge transfer

No.	Question	Answers to the questions asked in the survey	Respondents' answers	Average value	Median	Standard Deviation
1	What sources do you consider crucial for knowledge acquisition by employees in businesses?	Training courses provided by external trainers	114	72,5	78,5	28,54
		Training courses conducted by internal trainers	78			
		Knowledge exchange between staff	107			
		Contacts and cooperation with customers	79			
		The Internet	87			
		Knowledge repositories in businesses	26			
		Conferences, symposiums	50			
2	What factors influence eagerness to share knowledge?	Market analysis reports	39	78,44	82,00	24,90
		Trust towards employees	112			
		Respect for employees.	112			
		Organizational culture of the business	70			
		Appropriate motivation	88			
		Remuneration	95			
		Development and training courses	82			
3	Which statements apply to your company?	Organizational culture	39	55,25	47,00	18,59
		Friendly management staff	62			
		Communication channels	46			
		Sharing knowledge is a key process in any business.	77			
		Employees are encouraged to share knowledge.	74			
		Most employees exchange knowledge between one another.	83			
		I feel personal development through knowledge sharing.	50			
4	What tools are used to disseminate knowledge in your organization?	My supervisor notices me sharing knowledge with my coworkers.	42	53,83	56,50	15,82
		My supervisor is happy to share his knowledge with me.	44			
		The organization promotes a culture of knowledge sharing.	29			
		All employees are aware that knowledge sharing is a key element in the implementation of the company's strategy.	43			
		Knowledge repository	50			
		Interactive platforms	72			
		The employer's internal site Intranet	68			
5	What barriers in your opinion may exist to knowledge management in your organization?	Social media	63	51,11	38,00	28,08
		Forum	44			
		E-mail	74			
		Does not have any	26			
		High turnover among employees	64			
		Lack of openness of employees to share knowledge	86			
		Lack of trust	108			
		Common reluctance to share knowledge	53			
		Unadapted IT infrastructure	38			
		No existing communication channels	34			
6	What are the important benefits of knowledge management?	Lack of existing knowledge sharing tools	30	39,83	35,50	16,59
		Lack of funds	28			
		There are no barriers	19			
		Increased engagement and job satisfaction	53			
		Improving the information flow	67			
		Competitive edge	34			
		Better cooperation between organizational units	36			
6	What are the important benefits of knowledge management?	Learning from mistakes	35	39,83	35,50	16,59
		Better resource allocation	14			

Source: Own study

intranet websites) – 28.5%, and communication channels, which include all social media – 26.4%. Attention is drawn to the number of respondents whose employing businesses do not have knowledge dissemination tools implemented – 10.9%.

Barriers preventing smooth knowledge management in businesses are a very important element. Over half of the respondents believe that the lack of trust between employers and employees creates a barrier to the knowledge exchange – 45.2%. The lack of trust is also related to a lack of openness of employees to the knowledge-sharing process itself – 36%. In fact, one clearly results from the other. Trust and openness are closely related and determine the employees' attitudes. The barriers to knowledge management also include a high employee turnover and rotation rate, which is associated with the lack of knowledge transfer between employees – 26.8% and a widespread reluctance to share knowledge – 22.2%. Only 7.9% of respondents believe that there are no barriers affecting processes related to knowledge management within organizations. In terms of benefits, the highest scores were given to: better information and knowledge flow between employees (28%) and commitment and job satisfaction (22.2%) as non-financial motivators. Learning from mistakes (14.6%), as well as better cooperation between organizational units (15.1%) were also highly rated by the respondents.

Figure 4 presents the results of the respondents' answers to the question of what knowledge management model is used in the businesses in which they are employed. The resource model, which assumes that the most important factor in knowledge exchange in organizations is employees, is the most popular among the respondents (38.1%). Slightly fewer responses were obtained by the implemented process model (36.4%), which is characterized by dissemination of knowledge by the company itself serving as the initiator. Also highly rated was the Japanese model (25.5%), which assumes that each employee influences the company's state of knowledge.

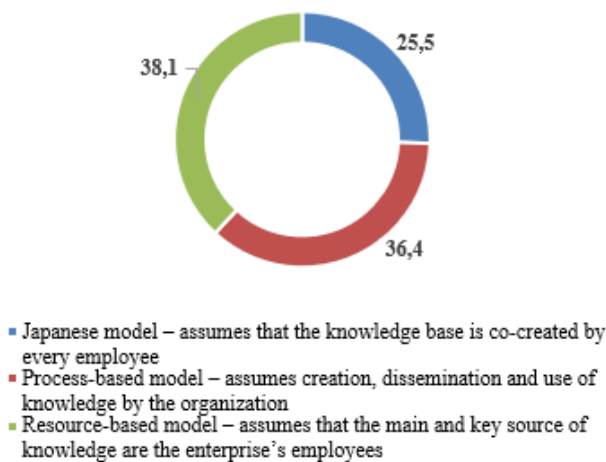


Fig. 4 Knowledge management models in businesses
Source: Own study

Figure 5 presents tools for knowledge development in enterprises. Most respondents take advantage of knowledge development through participation in trainings (57.3%), courses (37.2%) and conferences (20.5%). It seems that these forms of development are the most accessible to employees and can be easily utilized. Methods such as coaching (11.3%) or mentoring (7.9%) are much less used. Figure 6, on the other hand, shows the frequency of participation in selected forms of development presented in Figure 5. According to available data, in 2021-2022 almost 38.9% of respondents did not participate in any of the above-mentioned forms of development. Over 30% participated once or twice, while 17.2% participated 3 or 4 times. A small percentage (12.6%) is people, who actively sought ways of development in 2021-2022.

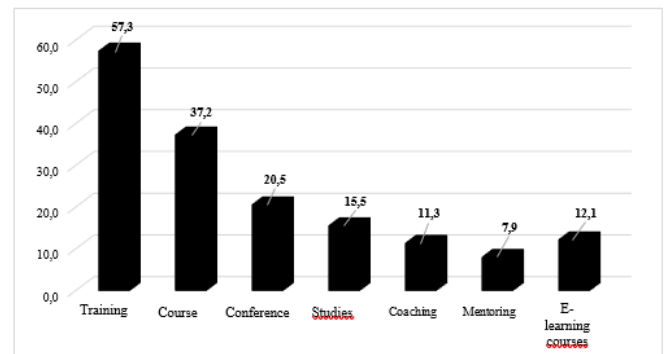


Fig. 5 Methods of knowledge development in businesses
Source: Own study

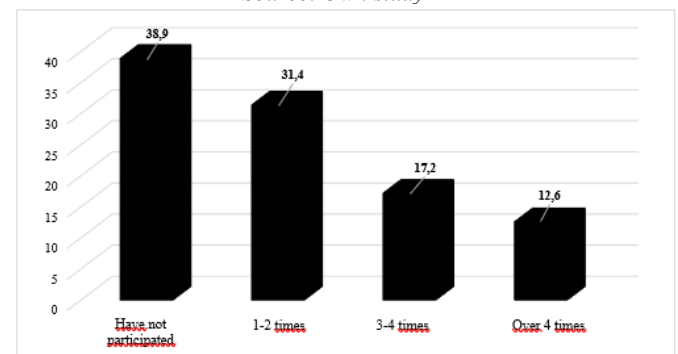


Fig. 6 Participation in selected forms of development
Source: Own study

V. RESULTS AND DISCUSSION

The conducted research showed that implementation of hybrid work models is still at an early stage. A significant percentage of respondents, who took part in the survey continue to perform their work in stationary models. Knowledge management in enterprises is a key process. A flexible and continuous flow of information enables employees to efficiently perform their tasks. Respondents notice the importance and necessity of knowledge sharing in favorable conditions. They pay particular attention to openness and trust on the part of employers. They easily identify the benefits of openness, as well as the barriers to knowledge transfers be-

tween colleagues. Digital tools play an important role both in the hybrid work model as well as knowledge management, which enable knowledge transfer in the hybrid model. Studies of this research sample proved that respondents notice the importance of programs such as MS Teams, ZOOM, or ClickMeeting as a basic tool used in performing their duties outside the employer's premises. They state their knowledge of the software at a moderately good level, which means that they use these tools only for occasional meetings. On the other hand, in the global perspective, the tools used to disseminate knowledge in organizations are primarily interactive platforms, which include LMS (Learning Management System) or e-learning platforms. Communication channels such as social media that are available to all employees are very popular. The tools used to popularize knowledge are not only IT programs, but also forms such as training, courses, conferences, studies, or more individualized forms, which include coaching or mentoring generally available to employees.

The results of the research conducted related to the adopted multi-variant remote work model in relation to knowledge management in businesses are as follows:

- employees use ICT tools to perform remote work in order to transfer knowledge. These include MS Teams, ZOOM, interactive platforms, social media and electronic devices,
- the respondents work in a stationary work model, while they expect a hybrid model,
- knowledge exchange between employees flows best in the employee-employee relationship and the manager-employee relationship,
- respondents notice many sources of knowledge management processes. These include: acquiring knowledge through training, sharing knowledge or acquiring it from customers,
- sharing is the first step towards the implementation and dissemination of a knowledge sharing culture.

According to a study by the House of Skills concerning 'Trends in human development for 2022' [32], the results indicate the further development and future of hybrid work in businesses. The challenge will lie in competent management of the dispersed team taking into account employees performing work from offices and homes in real time. However, there is a lack of clear guidance from companies on the use of hybrid work. According to the Gartner report [33], many companies have not yet developed internal strategies for using hybrid work. A study conducted by the Future Business Institute [34] shows that 75% of respondents believe that stationary work should be combined with remote work, i.e. a hybrid model of tasks performance by employees should be established. Merely 12% of respondents are of the opinion that they can only do their work remotely. However, 51% of respondents believe that performing work at home stimulates their innovativeness.

According to a recent survey by The Voice of the European Workforce [35], results show that employees believe that companies that have implemented knowledge transfer

are much more competitive in terms of customer satisfaction and contentment, as well as revenue growth. Employees perceive such companies as innovative and valuable. The future in the dissemination of knowledge lies in tools such as artificial intelligence, databases or natural language processing.

Data from the Future of Jobs Report [36] indicate technologies that can be implemented by businesses by 2025. These include:

- 87% encryption and cybersecurity,
- 86% artificial intelligence,
- 80% the possibility of using cloud tools,
- 73% extension of big data analytics,
- 71% e-commerce,
- 69% development of robotization (automation).

VI. CONCLUSION

In the 1960s, IT systems started being developed to support knowledge management processes in businesses. Their goal was to quickly and easily transfer knowledge between employees or create places where knowledge can be collected and processed. The larger the company's knowledge repository is, the greater its attractiveness on the market. This affects the number of applications sent in by people who want to be employed in a company with a well-established market position. Knowledge determines the decisions made by the management and directly influences strategic actions. Competent knowledge management and transfer also affect the work models in which employees work. Remote or hybrid work models, which combine remote and stationary work, are becoming increasingly common. It is a model that is expected by employees. It refers to maintaining a balance between professional and private life, limiting factors such as the time for commuting to work.

The first research hypothesis (H1) was positively verified: knowledge sharing significantly affects employees. The respondents believe that the knowledge sharing process is a fundamental element in establishing interpersonal relationships, building trust and openness. It is a process that should encourage companies to build a culture of knowledge sharing among employees. Respondents see a direct link and influence on building a company's competitiveness and market position.

The conducted research indicated that employees most often use the tools provided by their employer to transfer knowledge. They also use software intended for knowledge exchange and communicate between employees. These include: MS Teams, ZOOM, ClickMeeting, and many others. It is also worth noting that e-learning platforms or internal employer systems such as the intranet (H2) are used for knowledge exchange. Most respondents believe that performing tasks in a hybrid work model is an effective and efficient solution, while 44.8% of respondents continue to work in a stationary model. The hybrid model allows for individual work, but also the opportunity to meet the team live on days that are scheduled for presence at the workplace. The respondents have no objections to knowledge transfer

that is implemented in their organizations. The best flow of information is in the manager-employee relationship. This is due to the fact that the manager contacts their employee regarding: tasks performed, delegation of duties or providing organizational information. Employees also share information, experience, or various other insights with each other. Respondents value openness and trust on the part of the employer in terms of knowledge transfer. They consider this to be one of the key processes functioning in businesses. There are benefits such as: better information flow, increased engagement, and barriers: lack of trust or widespread reluctance to share knowledge.

The third hypothesis (H3) regarding the relationship between knowledge management and the multi-variant model of remote work adopted in the article was positively verified. According to the conducted research, the model created for the purposes of this article was reflected in the respondents' answers to the survey questions. Both knowledge creation, transfer and transformation are basic processes that affect the flexibility and reach of knowledge transfers. This is done by matching appropriate ICT tools that will correspond to the needs of employees. The direction of the conducted research indicates dissemination of a knowledge sharing culture in organizations. Knowledge is a multi-faceted element. It is not possible to easily determine its components. It also encompasses emotions, feelings, values. Important contributing factors include the workplace atmosphere and a sense of trust.

The study is limited by the fact of including a randomly selected research sample working in companies of all sizes and industries. Therefore, the study has an illustrative dimension.

In the perspective of further research, they should be carried out in specific companies in order to be able to analyze how such companies approach the hybrid model and knowledge management. Research should be carried out taking into account specific industries and professions performed by the respondents. This will allow a clear reference to the results of previous research.

REFERENCES

- [1] J. M. Nilles, *The telecommunications – transportation trade off: Options for tomorrow*, Wiley 1976, p. 87
- [2] J. M. Nilles, *Telework – Strategy of managing a virtual crew*, Wydawnictwo Naukowe – Techniczne, Warsaw 2003, p. 25
- [3] J. Wachowicz, *Virtual organizations – genesis, characteristics and advantages of Electronic Commerce – economy of the XXI century*, Wyd. MKN E-C, Gdańsk University of Technology, Gdańsk 2001, p. 125
- [4] T. Zalega, *Remote work – an illustration of changes in Poland and selected European Union countries*, "Master of Business Administration" 2009, No. 17(4) p. 35-45., p. 37
- [5] A. Jeran, *Remote work as a source of issues in the performance of work*, "Opuscula Sociologica" 2016, No. 2, p.50.
- [6] M. Hynes, *Telework Isn't Working: A Policy Review*, "The Economic and Social Review", Vol. 45, No. 4, 2014, p. 579-602.
- [7] B. Szluz, *Teleworking – a modern, flexible form of employment and work organization – a opportunity or risk?*, "Modern Management Review", No. 4, 2013, p. 254.
- [8] A. Greenberg, A. Nilssen, *WR Paper: Addressing 21st Century Challenges through Telework*, Duxbury: Wainhouse Research, 2008.
- [9] S. Ciupa S., *Employment of employees in the form of telework according to the Labor Code*, "Monitor Prawa Pracy", No. 12, 2007, p. 622-623
- [10] Act of 26 June 1974, art. 67, § 1 and 2. Labour Code
- [11] M. Carnoy, *Sustaining Flexibility: Work, Family and Community in the Information Age*. Cambridge: Harvard University Press, 2000
- [12] <https://www.wework.com/pl-PL/ideas/workspace-solutions/flexible-products/hybrid-workplace> [access date 01.04.2022]
- [13] <https://gojtowska.com/2020/07/01/praca-w-modelu-hybridowym/>
- [14] P. Drucker, *Pro-capitalist society*, Wydawnictwo Naukowe PWN, Warsaw 1999, p. 13
- [15] M. Klak, *Knowledge Management in a Contemporary Business*, Publishing House of the Prof. Edward Lipiński University of Economics and Law in Kielce, Kielce 2010, p. 13
- [16] B. Mikula, A. Pietruszka-Oryl, A. Potocki, *Managing a XXI Century Business*, Difin, Warsaw 2002, pp. 69-71
- [17] P. Drucker, *Pro-capitalist Society*, Wydawnictwo Naukowe PWN, Warsaw 1999, p. 43
- [18] E. Turban, *Expert Systems and Applied Artificial Intelligence*, Prentice Hall Collage, Macmillan 1992
- [19] S. Galata, *Strategic Information Management. Knowledge, intuition, strategy, ethics.*, Difin, Warsaw 2004, p. 50
- [20] T. M. Amabile, *A model of creativity and innovation in Organizations*, Research and Organizational Behaviour, 1988, Vol. 10
- [21] M. Juchnowicz, *Human capital management. Processes - Tools – Applications*, Polish Wydawnictwo Ekonomiczne, Warszawa 2014
- [22] N. W. Foote, E. Matos, N. Rudd, *Managing the knowledge manager*. "The McKinsey Quaterly", No. 3, 2001, p. 121
- [23] I. Nonaka, H. Takeuchi H., *The knowledge creating company: how Japanese companies create the dynamic of innovation*, Oxford University Press, New York, 1995, p. 284
- [24] K. Schwaber K., *Agile Project Management with Scrum*, "Microsoft Press", 2004, p. 56
- [25] F. L. Schmidt, J.E. Hunter, *Tacit knowledge, practical intelligence, general mental ability, and job knowledge*, "Current Directions in Psychological Science", 1993, p. 8-9
- [26] R. Ribeiro, H. Collins H., *The bread-making machine: Tacit knowledge and two types of action*, "Organ. Stud.", 28 (9), 2007, p. 1417-1433
- [27] D. J. Skyrme, *Knowledge Networking. Creating the Collaborative Enterprise*, Butterworth Heinemann, Oxford, 1999, p. 51-59
- [28] N. N. Sunassee, D.A. Sewry D. A., *A Theoretical Framework for Knowledge Management Implementation*, ACM International Conference Proceeding Series, Port Elizabeth, South Africa, 2002, p. 235-245
- [29] J. Barney J., *Firm resources and sustained competitive advantage*, "J. Management", 77 (1), 1991, p. 99-120
- [30] I. Nonaka I., *The knowledge-creating company*, "Harvard Business Review", 69 (6), 1991, p. 96-104
- [31] E. Krok, *Social media as an element of the knowledge management system in a company*, Scientific Journals of the University of Szczecin, Studia Informatica No. 28, No. 656, 2011, pp. 53-54
- [32] <https://www.houseofskills.pl/wp-content/uploads/2022/01/Trendy-w-rozwoju-ludzi-na-rok-2022.pdf> [access date 01.04.2022]
- [33] <https://emtemp.gcom.cloud/ngw/globalassets/en/human-resources/documents/trends/top-priorities-for-hr-leaders-2022.pdf> [access date 01.04.2022]
- [34] <https://futurebusiness.institute/> [access date 01.04.2022]
- [35] <https://www2.deloitte.com/pl/pl/pages/human-capital/articles/raport-The-voice-of-the-European-workforce-2020.html> [access date 01.04.2022]
- [36] https://www3.weforum.org/docs/WEF_Future_of_Jobs_2020.pdf [access date 01.04.2022]

Named Entity Recognition System for the Biomedical Domain

Raghav Sharma
 Indian Institute of Technology Kharagpur
 Kharagpur, India
 Email: sharmaraghav.20@iitkgp.ac.in

Deependra Chauhan, Raksha Sharma
 Indian Institute of Technology Roorkee
 Roorkee, India
 Email: {d_chouhan, raksha.sharma}@cs.iitr.ac.in

Abstract—The recent advancements in medical science have caused a considerable acceleration in the rate at which new information is being published. The MEDLINE database is growing at 500,000 new citations each year. As a result of this exponential increase, it is not easy to manually keep up with this increasing swell of information. Thus, there is a need for automatic information extraction systems to retrieve and organize information in the biomedical domain. Biomedical Named Entity Recognition is one such fundamental information extraction task, leading to significant information management goals in the biomedical domain. Due to the complex vocabulary (*e.g.*, *mRNA*) and free nomenclature (*e.g.*, *IL2*), identifying named entities in the biomedical domain is more challenging than any other domain, hence requires special attention. In this paper, we deploy two novel bi-directional encoder-based systems, *viz.*, BioBERT and RoBERTa to identify named entities in the biomedical text. Due to the domain-specific training of BioBERT, it gives reasonably good performance for the NER task in the biomedical domain. However, the structure of RoBERTa makes it more suitable for the task. We obtain a significant improvement in F-score by RoBERTa over BioBERT. In addition, we present a comparative study on training loss attained with ADAM and LAMB optimizers.

I. INTRODUCTION

INFORMATION extraction in the biomedical domain involves the identification of the independent pieces of information, for example, cause-effect arguments, causal triggers, adverse drug reaction, *etc.* Automated extraction of information from the biomedical text is an essential facilitator of clinical research and informed diagnosis [1], [2]. The presence of many domain-specific terminologies in biomedical literature makes information extraction a challenging task.

An entity is a word or a sequence of words in the text with a physical existence with different properties. Named entity recognition (NER) is a sub-task of information extraction that seeks to identify and classify named entities as predefined categories in unstructured text. NER always serves as the foundation for many natural language applications such as *question answering*, *text summarization*, and *machine translation*. Biomedical Named Entity Recognition (BioNER) is a task of identifying biomedical named entities such as *gene*, *disease*, *drug*, *species*, *etc.*, in the raw text. Because of the complexity of biomedical nomenclature, BioNER is a more challenging task than NER in general. A gene name often contains a mix of alphabet, digits, hyphens, and other characters, for example, *HIV-1*. The domain frequently uses

abbreviations (“IL2” for “Interleukin 2”). In addition, the same biomedical named entities can be expressed in various forms. For example, gene names often contain alphabets, digits, hyphens, and other characters, thus having many variants (*e.g.*, “HIV-1 enhancer” versus “HIV 1 enhancer”). Moreover, many abbreviations (*e.g.*, “IL2” for “Interleukin 2”) have been used for biomedical named entities. Sometimes, the same entity can have very different aliases (*e.g.*, “PTEN” and “MMAC1” refer to the same gene) [1]. Another challenge of BioNER is the ambiguity problem. The same word or phrase can refer to more than one type of entities or does not refer to an entity depending on the context (*e.g.*, “TNF alpha” can refer to a protein or DNA).

Table I shows a few example sentences from the biomedical domain with the named entities and their types. Named Entity Recognition in the biomedical domain has been tried using various available methodologies and continues to be an active research topic due to the complexity and utility of the problem. BioBERT [3] is a language model trained on biomedical data to produce distributed representation of words. This paper presents a deep neural system for named entity recognition in the biomedical domain using BioBERT. Specifically, in this paper, we deploy two novel bi-directional encoder-based systems, *viz.*, BioBERT and RoBERTa to identify named entities in the biomedical text. Due to the domain-specific training of BioBERT, it gives reasonably good performance for the NER task in the biomedical domain. However, the supportive structure of RoBERTa makes it more suitable for the BioNER task than BioBERT.

II. RELATED WORK

Named Entity Recognition in the biomedical domain is a fundamental text mining task. It has attracted a lot of attention from researchers across different languages. Methodologies applied to this problem range from the traditional rule-based approaches to the most recent deep learning models. Due to the non-standard use of abbreviations, synonyms, synchronizations, ambiguities, and the frequent use of phrases to describe the entities, NER in the biomedical domain is still a challenging task [4].

Rule-based methods rely on hand-crafted rules to identify and classify named entities in text. An exhaustive lexicon almost always boosts the performance of these models. NER

TABLE I
EXAMPLES OF SENTENCES WITH THE NAMED ENTITY ANNOTATIONS

S.No.	Sentences with Annotations
1	Identification of APC2, a homologue of the {adenomatous polyposis coli tumour}_gene suppressor.
2	{Methanoregula formicica}_species sp.nov., a methane-producing archaeon isolated from methanogenic sludge.
3	{IL-2}_gene gene expression and {NF-kappa B}_protein activation through {CD2B}_antibody requires reactive oxygen production by {5-lipoxygenase}_protein.
4	Assymetrical cell division was observed in rod-shaped cells.

tools in the Biomedical domain rely on specific features to capture the characteristics of the different entity classes until recently. For instance, the suffix *-ase* is more frequent in protein names than in diseases; species names often consist of two tokens and have Latin suffixes; chemicals often contain specific syllabi like *methyl* or *carboxyl* [5]. However, hand-crafted semantic and syntactic rules often make these models data specific. Any change in the source of data will drop the performance of the system [5], [6]. As a result, rule-based approaches lead to a high precision but low recall.

Advancements in supervised machine learning were also applied to generic NER. NER can be considered like a multi-class classification or sequence labeling task. The correct selection and engineering of features are vital to the model's performance based on them. Many machine learning models have been tried and researched based on these features. These include Hidden Markov Models (HMMs) [7], decision trees [8], SVMs [9] and Conditional Random Fields (CRFs). A major requirement for supervised machine learning models to perform well is the presence of sufficient labeled/structured data. However, the presence of labeled data is limited, leading to the rise of unsupervised learning approaches. These models tend to focus more on corpus statistics (e.g. IDF), terminologies, and syntactic knowledge KALM [10].

More recently, deep learning methods that can automatically develop and extract features from the raw text are used end-to-end for generic NER. These models generally use character or word-level embeddings such as Word2Vec and GloVe as their basic input. Various models based on CNNs and RNNs have been researched. However, the BiLSTM-CRF model [11] has been most commonly used. Transformer-based models [12] have proven to be superior in quality and also take less time to train. Based on transformers, several pre-trained language models have been released, which on fine-tuning give state-of-the-art performance on various end tasks. These include Generative Pre-trained Transformer (GPT) [13] (left to right architecture) and Bidirectional Encoder Representations from Transformers (BERT) [14] (takes both left and right context). Bio-BERT shows that pre-training BERT on biomedical data significantly improves its performance on end tasks in the biomedical domain. This paper uses BioBERT for named entity recognition in the biomedical domain. However, BioBERT takes a significant amount of time to train; we reduce the training time of BioBERT. We also modify the pre-training settings of BioBERT, which enables us to achieve

TABLE II
STATISTICS OF BIOMEDICAL NER DATASETS

Dataset	Entity Type	No. of annotations
NCBI-Disease [16]	Disease	6881
BC5CDR [17]	Drug/Chem	15411
BC2GM [18]	Gene/Protein	20703
Species-800 [19]	Species	3708

better performance on the end task, that is, Named Entity Recognition in the biomedical text.

III. DATASET

We preprocess the four datasets in the biomedical domain, *viz.*, NCBI-Disease, BC5CDR (drug/chem, disease), BC2GM, and Species-800. The preprocessing of the NCBI-Disease dataset results in fewer annotations than the original dataset because duplicate articles are removed from its training set. The *Species-800* dataset was preprocessed and split as per Pyysalo et al., [15]. The statistics of the biomedical NER dataset are listed in Table II.

IV. METHODOLOGY

This paper presents a deep architecture for the named entity recognition in the biomedical domain. Our system deploys the representations of the words by a domain-specific language model, that is, BioBERT. We further optimize the system for the task using the LAMB optimizer. Furthermore, RoBERTa model is built on top of the BERT model. The architecture similarity with the BERT model makes RoBERTa model suitable for the named entity recognition task. This section describes the algorithm and its components.

A. BioBERT

Text documents in the biomedical domain contain a considerable amount of domain-specific proper nouns, (*e.g.*, *BRACA1*), which requires expertise in the domain to understand named entities. The general-purpose language representation models such as GloVe and Word2Vec give a poor performance for biomedical texts [20], [21]. The distribution of the words shifts from general domain corpora to biomedical corpora; hence direct application of generic word embeddings results in unsatisfactory performance [5], [15], [22]. BioBERT (Bidirectional Encoder Representations from Transformers for Biomedical Text Mining) is trained on the biomedical corpus. First, BioBERT is initialized with weights from BERT; BERT

was pre-trained on general domain corpus to overcome the data sparsity problem and bring more coverage. The main advantage of BERT over previous language model approaches like combinations of LSTMs and CRF is that BERT has relatively a simple architecture based on bidirectional transformers. Based on its last layer representations, BERT computes only token level probabilities in the BIOES format (Begin, Inside, Others). Then BioBERT is trained on biomedical corpora from PubMed. BioBERT is the first domain-specific BERT model which has been trained for biomedical-specific tasks [22].

B. RoBERTa

RoBERTa stands for Robustly Optimized BERT Pre-training Approach. Most of the training procedure of BERT and RoBERTa is common. However, there are a few fundamental structural differences. This section presents the differences between the two models.

1) *Static Masking vs. Dynamic Masking*: BERT relies on randomly masking and predicting tokens. In the original BERT model, each sequence was masked in only ten different ways over 40 epochs. RoBERTa uses dynamic masking in which different mask is generated every time a sequence is fed to the model.

2) *Model Input Format and Next Sentence Prediction*: The BERT model is also trained for the Next Sentence Prediction (NSP) objective along with the masked language modeling objective. The objective of auxiliary Next Sentence Prediction loss is to determine whether the segments belong to the same or different documents. It is equiprobable for document segments to be sampled continuously from the same or distinct documents. RoBERTa, on the other hand, takes a different approach by ignoring the NSP loss. The input representation can be seen as packed with full sentences sampled contiguously from one or more than one documents. The maximum input length is set to 512 tokens.

3) *Training with Large Batches*: The same computational cost models can be made by increasing the batch size and decreasing the number of steps. The original BERT was trained with 256 sequence batch size for 1 million steps via gradient accumulation. This computational cost can approximate the training model for 125k steps with a batch size of 2k or 31k steps for 8k. It can be inferred from previous works done on neural networks that training the model with large mini-batches improves end-to-end performance. RoBERTa uses a batch size of 8k.

4) *Text Encoding*: The difference between the BPE vocabulary of original BERT and RoBERTa lies in the sub-word size, preprocessing of input, and tokenization rule. BERT uses 30k, whereas RoBERTa uses a more extensive vocabulary of 50k subwords. BERT does preprocess of input while RoBERTa expands vocabulary size without additional preprocessing. RoBERTa raises vocabulary size without other tokenization rules.

C. LAMB Optimizer

This paper also shows the efficacy of the Large Batch Optimization (LAMB) algorithm with BioBERT for NER in

TABLE III
BIOBERT TOKEN LEVEL EVALUATION WITH ADAM OPTIMIZER

Dataset	Precision	Recall	F-score	Loss
NCBI Disease	88.8	91.8	90.2	33.71
BC5CDR	89.2	90.5	89.9	37.56
BC2GM	88.7	89.4	89.0	37.56
Species-800	79.1	83.2	81.1	32.89

TABLE IV
BIOBERT TOKEN LEVEL EVALUATION WITH LAMB OPTIMIZER

Dataset	Precision	Recall	F-score	Loss
NCBI Disease	86.9	91.8	90.2	11.26
BC5CDR	89.2	90.5	89.9	10.74
BC2GM	88.7	89.4	88.88	17.17
Species-800	79.1	83.2	80.28	12.52

the biomedical domain. LAMB helps to reduce the training time, and boost performance for text processing task [23]. Large batch training is the key to reducing deep neural networks' training time in a large distributed system. LAMB is a layer-wise adaptive large batch optimization technique. The generalization gap becomes a problem in the case of training large batches models. If direct optimization is performed, it may cause performance degradation. Devlin et al., [24] implemented BERT with a variant of ADAM optimizer, which uses ADAMs optimizer along with weight decay for training. LARS is another successful adaptive optimizer that has been used for large batch convolutional neural networks, but they are not effective for text processing tasks [23]. LAMB has shown superior performance across BERT and ResNet-50 training tasks with minimal hyperparameter tuning. Hence, we train BioBERT with the LAMB optimizer to optimize the training time. In addition, we show the superiority of the LAMB optimizer over the ADAM optimizer for the BioNER task (Section VI).

TABLE V
BIOBERT ENTITY LEVEL EVALUATION

Dataset	Precision	Recall	F-score
NCBI Disease	86.92	89.27	88.08
BC5CDR	92.74	92.79	92.77
BC2GM	83.59	83.39	83.74
Species-800	71.39	76.79	73.99

TABLE VI
ROBERTA ENTITY LEVEL EVALUATION

Dataset	Precision	Recall	F-score
NCBI Disease	87.32	88.84	88.07
BC5CDR	93.59	92.95	93.27
BC2GM	93.41	92.16	92.78
Species-800	76.01	84.00	79.81

V. EXPERIMENTAL SETUP

The overall process can be divided into pre-training and fine-tuning BioBERT. The pre-training weights are taken from Cohen and Hunter [2]. The fine-tuning step is problem-specific. For example, the model needs to be fine-tuned for named entity recognition, relation extraction, question-answering, and tasks independently. We fine-tune the BioBERT model for our dataset's named entity recognition task. A batch size of 8 was chosen for fine-tuning. The learning rate was set to $1e-5$, and the model was trained for 10 epochs. F-score is computed at the token level, word level, and entity level, that is, phrase level.

VI. RESULTS

Results are focused on two aspects: the optimizer's performance during training and the F-score for the task. We compare the training Loss by ADAM optimizer and LAMB optimizer. ADAM optimizer is a frequently used optimizer for the classification task. Table III and Table IV present the F-Score obtained with BioBERT at token level with ADAM and LAMB respectively. The last column of Table III and Table IV shows the Loss attained during training with ADAM and LAMB, respectively. We observed a significant difference in the training Loss value with the LAMB optimizer; hence LAMB made the model converge in a significantly shorter time than ADAM. However, there is no significant difference in the F-score obtained with ADAM and LAMB. Table V and Table VI show the comparison between BioBERT and RoBERTa for NER across the four datasets from biomedical domain. We fine-tune both the models on our dataset for the named-entity recognition task. RoBERTa significantly improved the F-score for the named-entity recognition in the biomedical domain.

VII. CONCLUSION

Named entity recognition in the biomedical domain is a challenging task considering the unconstrained nomenclature of the biomedical vocabulary. This paper presents a named-entity recognition system for the biomedical domain. We deploy two pre-trained language models for the task, *viz.*, BioBERT and RoBERTa. Due to the domain-specific training of BioBERT, it gives reasonably good performance for the NER task in the biomedical domain. However, the structure of RoBERTa makes it more suitable for the task. Simple fine-tuning of RoBERTa on the dataset for BioNER boosts the results significantly. Additionally, we show a comparison between the training loss attained with ADAM and LAMB optimizers.

REFERENCES

- [1] S. Ananiadou and J. McNaught, *Text mining for biology and biomedicine*. Citeseer, 2006.
- [2] K. B. Cohen and L. Hunter, "Getting started in text mining," *PLoS computational biology*, vol. 4, no. 1, 2008.
- [3] J. Lee, W. Yoon, and S. Kim, "Biobert: a pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics*, 2019.
- [4] U. Leser and J. Hakenberg, "What makes a gene name? named entity recognition in the biomedical literature," *Briefings in Bioinformatics*, vol. 6, no. 4, p. 357–369, 2005.
- [5] M. Habibi, L. Weber, M. Neves, D. L. Wiegandt, and U. Leser, "Deep learning with word embeddings improves biomedical named entity recognition," *Bioinformatics*, vol. 33, no. 14, pp. i37–i48, 2017.
- [6] S. Eltyeb and N. Salim, "Chemical named entities recognition: a review on approaches and applications," *Journal of cheminformatics*, vol. 6, no. 1, pp. 1–12, 2014.
- [7] D. M. Bikel, R. Schwartz, and R. M. Weischedel, "An algorithm that learns what's in a name," *Machine Learning*, vol. 34, no. 1-3, pp. 211–231, 1999.
- [8] G. Szarvas, R. Farkas, and A. Kocsor, "A multilingual named entity recognition system using boosting and c4.5 decision tree learning algorithms," *Discovery Science Lecture Notes in Computer Science*, p. 267–278, 2006.
- [9] P. McNamee and J. Mayfield, "Entity extraction without language-specific resources," *proceeding of the 6th conference on Natural language learning - COLING-02*, 2002.
- [10] A. Liu, J. Du, and V. Stoyanov, "Knowledge-augmented language model and its application to unsupervised named-entity recognition," *Proceedings of the 2019 Conference of the North*, 2019.
- [11] Z. Huang, W. Xu, and K. Yu, "Bidirectional lstm-crf models for sequence tagging," 2015, cite arxiv:1508.01991. [Online]. Available: <http://arxiv.org/abs/1508.01991>
- [12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.
- [13] A. Radford and K. Narasimhan, "Improving language understanding by generative pre-training," 2018.
- [14] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *CoRR*, vol. abs/1810.04805, 2018. [Online]. Available: <http://arxiv.org/abs/1810.04805>
- [15] S. Pyysalo, F. Ginter, H. Moen, T. Salakoski, and S. Ananiadou, "Distributional semantics resources for biomedical text processing in proceedings of Ibm. 2013," *Google Scholar*, pp. 39–44, 2013.
- [16] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *EMNLP*, vol. 14, 2014, pp. 1532–1543.
- [17] J. P. C. Chiu and E. Nichols, "Named entity recognition with bidirectional lstm-cnns," 2015, cite arxiv:1511.08308Comment: To appear in Transactions of the Association for Computational Linguistics. [Online]. Available: <http://arxiv.org/abs/1511.08308>
- [18] Z. Yang, D. Yang, C. Dyer, X. He, A. J. Smola, and E. H. Hovy, "Hierarchical attention networks for document classification," in *HLT-NAACL*, K. Knight, A. Nenkova, and O. Rambow, Eds. The Association for Computational Linguistics, 2016, pp. 1480–1489. [Online]. Available: <http://dblp.uni-trier.de/db/conf/naacl/naacl2016.html#YangYDHS16>
- [19] Ö. Uzuner, B. R. South, S. Shen, and S. L. DuVall, "2010 i2b2/va challenge on concepts, assertions, and relations in clinical text," *Journal of the American Medical Informatics Association*, vol. 18, no. 5, pp. 552–556, 2011.
- [20] S. Pawar, R. Sharma, G. K. Palshikar, P. Bhattacharyya, and V. Varma, "Cause-effect relation extraction from documents in metallurgy and materials science," *Transactions of the Indian Institute of Metals*, vol. 72, no. 8, pp. 2209–2217, 2019.
- [21] R. Sharma and G. Palshikar, "Virus causes flu: Identifying causality in the biomedical domain using an ensemble approach with target-specific semantic embeddings," in *International Conference on Applications of Natural Language to Information Systems*. Springer, 2021, pp. 93–104.
- [22] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang, "Biobert: a pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics*, vol. 36, no. 4, pp. 1234–1240, 2020.
- [23] Y. You, J. Li, S. Reddi, J. Hseu, S. Kumar, S. Bhojanapalli, X. Song, J. Demmel, K. Keutzer, and C.-J. Hsieh, "Large batch optimization for deep learning: Training bert in 76 minutes," 2020.
- [24] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

Representing and Managing Experiential Knowledge with Decisional DNA and its Drimos® Extension

Edward Szczerbicki
 Gdansk University of Technology
 ul. Narutowicza 11, 80233 Gdansk,
 Poland
 Email: esz@zie.pg.gda.pl

Cesar Sanin
 Australian Institute of Higher
 Education, 545 Kent St, Sydney,
 NSW 2000, Australia
 Email: c.sanin@aih.nsw.edu.au

Karina Sterling-Zuluaga
 Idream Technology Pty Ltd
 Sydney, NSW 2000, Australia
 Email: karina.sterling@drimos.ai

Abstract— The Semantic Web concept is proposing a future concept of the WorldWideWeb (WWW) where both humans and man-made systems are able to interconnect and exchange knowledge. One of the challenges of Semantic Web is smart and trusted accommodation of knowledge in artificial systems so it can be unified, enhanced, reused, shared, communicated and distributed with added aptitude. Our research represents an important component of addressing the above challenge and exciting, cutting-edge exploration trend in the general area of developing tool for intelligence augmentation.

I. INTRODUCTION

MOST experts agree that intelligent non-natural system is yet to be established. The main issue that remains a challenge is securing trust and explainability in such systems. This is where the notion of augmented intelligence comes into play. It is an alternate insight of artificial intelligence (AI) that focuses on AI's enhancing role [1]. Enhancing role of intelligence amplification inspired our initial research idea and vision to develop, expand, and extend an artificial intelligence augmentation system, an architecture that would support smart discovering, capture, adding, storing, improving and sharing information and knowledge among agents, machines, and organizations through experience. Bio-inspiration comes in this case from the fact that in nature experiences that all living organisms (including humans) go through during their operating lives support sustainable development, evolution, and add smartness to all associated functionalities and practices. The significance of experience in biological development cannot be overemphasized. We propose an original experience-based Knowledge Representation (KR) approach in which experiential knowledge is represented by Set of Experience (SOE) and is conveyed into the upcoming by Decisional DNA (DDNA) [2,3].

For the sake of completeness, SOE and DDNA are very briefly introduced here. Set of Experience Knowledge Structure (SOEKS) is a knowledge portrayal structure created to

acquire and store formal decision events in a organized and unambiguous way. It is composed by 4 fundamental elements: variables, functions, constraints, and rules. Variables are commonly used to represent knowledge in an attribute-value form, following the conventional approach for knowledge representation. Functions, Constraints, and Rules of SOEKS are techniques of associating variables. Functions define relationships between a set of input variables and a dependent variable; thus, SOEKS uses functions as a way to create links among variables and to build multi-objective purposes. Constraints are functions that act as a way to limit options, limit the set of possible results, and manage the performance of the system in relation to its aims. Lastly, rules are relationships that operate in the world of variables and express the condition-consequence connection as “if-then-else” and are used to represent inferences and partner actions with the conditions under which they should be executed [3]. Rules are also methods of recording specialist-defined knowledge into the system. The Decisional DNA is a edifice capable of capturing decisional characteristics of an individual or organization and has the SOEKS as its foundation. Several Sets of Experience can be assembled, classified, organized and sorted into decisional chromosomes, which mount up decisional policies for a specific region of decision-making occurrences. The set of chromosomes embrace, lastly, what is called the Decisional DNA (DDNA) [4].

II. DDNA KNOWLEDGE MANAGEMENT IN PRACTICE

DDNA technology has been verified, tested, and applied through numerous real life case studies and implementations. Table I below lists some of the most successful DDNA based real life applications with their references for possible further reading. All of them use our advanced portable and domain independent software representation for SOE and Decisional DNA embedded in DDNA Manager [4].

the treatment of experience in the buildup of the system's meta-knowledge.

D. Idream.Technology: From Individual to Collective Experiential Knowledge Management

The new and the furthestmost large-scale DDNA extension is the drimos® platform from Idream Technology Pty Ltd (www.drimos.ai) (Fig. 2). We have developed social digital platform using collective experience. This DDNA-based application, which commences in mid-2022 after comprehensive one year long design, testing, and validation, projects personalized road-maps to achieve purposes, goals, and aims taking into account individuals' personalities and circumstances. Specific areas of human activities covered within the platform include travel, education, acquisition, and well-being.

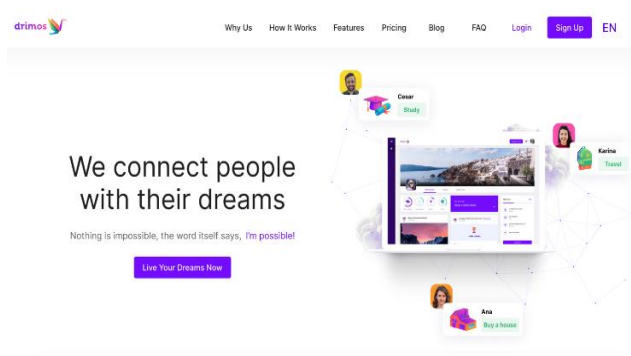


Fig. 2 The AI coaching platform to achieve your goals and dreams – drimos.ai homepage screenshot (from Idream Technology)

The popular existing social platforms personalize members' profiles classifying them by purchasing patterns or consumption of web content. drimos® platform from Idream Technology goes further because it mixes human intelligence with artificial intelligence, identifying how people make decisions. It captures, integrates, stores and reuses thousands of experiences occurring during the process of achieving personal, individual dreams or goals. In other words, it uses collective decision-making experience and applies it to amplify the individual one. As the result, dreamer's profile is presented with a systematic procedure to follow in order to achieve their personal dream or objective.

In terms of augmenting the human intelligence, drimos® considers two important elements to achieve its purpose: (i) a goal setting technique developed by an international expert in this area Karina Sterling (https://www.ted.com/talks/karina_sterling_cumple_tu_sue_no_y_cambia_el_mundo) which strengthens the emotional attachment to goals, and (ii) the capture of day-to-day anonymous experience from dreams/goals achievers which is explicitly formalized into Sets of Experience and added to the drimos® DDNA.

The main technique behind the process of managing experiential knowledge stored in the DDNA, is the

similarity concept based on mathematical distance between Sets of Experience [2,4]. This concept has been successfully used in all applications presented in Table I. In drimos® platform from Idream Technology the similarity notion is applied to the distance among users profiles of the drimos® community. The most similar profiles are chosen by the system to assist in the creation of successful path to reach the goals set up by others. Fig. 3 shows the similarity engine in action, and Fig. 4 presents the screenshot of gamification and set of tasks suggested to the user by the platform based on profiles' similarity.



Fig. 3 Similarity selection

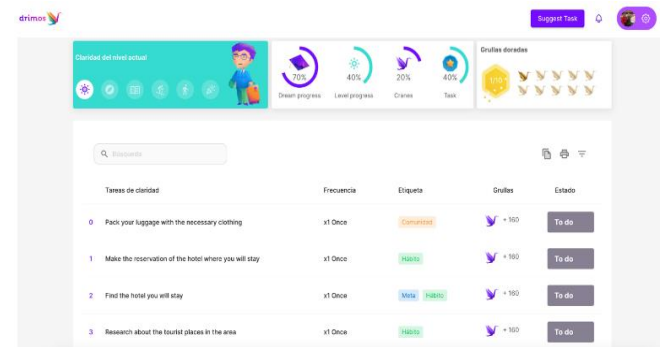


Fig. 4 Screenshot of goal steps with indicators and gamification model

Idream.Technology is a start-up hi-tech company to offer its smart enhancement of dedicated coaching services through drimos® application. It provides DDNA technology based tools to enhance peoples' sustainable development. It addresses global societies by covering English and Spanish speaking regions of the world.

IV. FUTURE OUTLOOK

Set of Experience (SOE) and Decisional DNA (DDNA) concepts are at the beginning of their advance but are already making a difference to knowledge management theory and practice. Future integration of various DDNA extensions would provide an intelligent and sustainable Internet application environment that would enable cybernetic positions (instruments that expediate interoperation among users, applications, and resources) to effectively capture, publish, share and manage explicit knowledge means and

sources. It would also provide support for on-demand services creating vast commercial potential for this technology. Through our approach to knowledge representation and formalization embedded in the concept of Decisional DNA, the future DDNA-based knowledge grid would incorporate epistemology and ontology to reflect human cognition characteristics and adopt the techniques and standards developed during work toward the next-generation, beyond-semantic web.

REFERENCES

- [1] N. Pathak, *The Future of AI*. Apress, Berkeley, CA, 2017, pp. 247–259.
- [2] C. Sanin, E. Szczerbicki, “Experience-based Knowledge Representation SOEKS”. *Cybernet Sys.* 40(2), 99-122, 2009.
- [3] C. Sanin, L. Mancilla-Amaya, Z. Haoxi and E. Szczerbicki, “Decisional DNA: The Concept and its Implementation Platforms”, *Cybernetic sand Systems*, 43:2, 67-80, 2012, DOI: 10.1080/01969722.2012.654069
- [4] E. Szczerbicki, C. Sanin, *Knowledge Management and Engineering with Decisional DNA*, Springer-Verlag, Berlin, 2020 DOI:10.1007/978-3-030-39601-5.
- [5] S. I. Shafiq, C. Sanin, C. Toro, and E. Szczerbicki, “Virtual engineering process (VEP): a knowledge representation approach for building bio-inspired distributed manufacturing DNA”, *International Journal of Production Research*, 54:23, 7129-7142, 2016, DOI: 10.1080/00207543.2015.1125545
- [6] C. Sanin, L. Mancilla-Amaya, E. Szczerbicki, and P. CayfordHowell, (2009) “Application of a Multi-domain Knowledge Structure: The Decisional DNA”, in *Intelligent Systems for Knowledge Management*, N. T. Nguyen, E. Szczerbicki editors: Springer Berlin / Heidelberg, Vol. 252, DOI 10.1007/978-3-642-04170-9_3
- [7] B. Kucharski, E. Szczerbicki, “Experience database based on a workflow class system”, *Foundations of Control and Management Science*, no 12, 2009.
- [8] H. Zhang, C. Sanin, and E. Szczerbicki, “Decisional DNA-based embedded systems: A new perspective”, *Systems Science*, Vol. 36, 2010.
- [9] L. Mancilla-Amaya, E. Szczerbicki, and C. Sanin, “A proposal for a knowledge market based on quantity and quality of knowledge”, *Cybernetics and Systems.* 44(2-13), 2013, DOI: 10.1080/01969722.2013.762233
- [10] M. M. Waris, C. Sanin, and E. Szczerbicki, (2019) “Establishing Intelligent Enterprise through Community of Practice for Product Innovation”, *Journal of Intelligent and Fuzzy Systems*, 2019 <http://dx.doi.org/10.3233/JIFS-179329>
- [11] B. Kucharski, E. Szczerbicki, “An approach to smart experience management”, *Cybernetics and Systems.* Vol. 42, 2011, DOI 10.1080/01969722.2011.541215
- [12] H. B. Jabrouni, G. Kamsu-Foguem, and C. Vaysse, “Continuous improvement through knowledge-guided analysis in experience feedback”, *Engineering Applications of Artificial Intelligence* 24(8), 2011.
- [13] H. Zhang, C. Sanin, and E. Szczerbicki, “Implementing fuzzy logic to generate user profile in decisional DNA television: The concept and initial case study”, *Cybernetics and Systems* 44(2-3), 2013, DOI: 10.1080/01969722.2013.762280
- [14] C. Toro, E. Sanchez, E. Carrasco, L. Mancilla-Amaya, C. Sanin, E. Szczerbicki, M. Graña, P. Bonachela, C. Parra, and G. Bueno, “Using set of experience knowledge structure to extend a rule set of clinical decision support system for Alzheimer’s disease diagnosis”, *Cybernetics and Systems*, 43(2), 2013, DOI: 10.1016/j.procs.2014.08.141
- [15] B. A. Muhammad, S.I. Shafiq, C. Sanin, and E. Szczerbicki, “Towards Experience-Based Smart Product Design for Industry 4.0”, *Cybernetics and Systems*, 50:2, 165-175, 2019, DOI: 10.1080/01969722.2019.1565123
- [16] M. M. Waris, C. Sanin, and E. Szczerbicki, “Toward Smart Innovation Engineering: Decisional DNA-Based Conceptual Approach”, *Cybernetics and Systems*, 47:1-2, 149-159, 2016, DOI: 10.1080/01969722.2016.1128775
- [17] T. de Souza Alves, de Oliveira C.S., C. Sanin, and E. Szczerbicki, (2018), “Knowledge-based Vision Systems: A Review”, Proceedings of Knowledge-Based Intelligent Information and Engineering Systems 22nd International Conference KES 2018, in *Advances in Knowledge-Based and Intelligent Information and Engineering Systems*: R. J. Howlett, L. C. Jain (Eds.), Belgrade, Sep 2018, Elsevier Procedia Computer Science, 2018, DOI: 10.1016/j.procs.2018.08.077
- [18] H. Zhang, F. Li, J. Wang, and E. Szczerbicki, “A Novel IoT-Perceptive Human Activity Recognition (HAR) Approach Using Multi-Head Convolutional Attention”, *IEEE Internet of Things Journal*, 7, 2019, DOI: 10.1109/jiot.2019.2949715
- [19] de Oliveira, C. S., C. Sanin, and E. Szczerbicki, (2019). “Towards Knowledge Formalization and Sharing in a Cognitive Vision Platform for Hazard Control (CVP-HC)”. *Proceedings Asian Conference on Intelligent Information and Database Systems* (pp. 53-61). Springer, Cham, 2019, DOI: 10.1016/j.procs.2020.09.179
- [20] C. Zanni-Merk, E. Szczerbicki, “Building collective intelligence through experience: the KREM model”, *Journal of Intelligent and Fuzzy Systems*, DOI: 10.3233/JIFS-179327, 2019.

Software, System and Service Engineering

THE S3E track emphasizes the issues relevant to developing and maintaining software systems that behave reliably, efficiently and effectively. This track investigates both established traditional approaches and modern emerging approaches to large software production and evolution.

For decades, it is still an open question in software industry, how to provide fast and effective software process and software services, and how to come to the software systems, embedded systems, autonomous systems, or cyber-physical systems that will address the open issue of supporting information management process in many, particularly complex organization systems. Even more, it is a hot issue how to provide a synergy between systems in common and software services as mandatory component of each modern organization, particularly in terms of IoT, Big Data, and Industry 4.0 paradigms.

In recent years, we are the witnesses of great movements in the area of software, system and service engineering (S3E). Such movements are both of technological and methodological nature. By this, today we have a huge selection of various technologies, tools, and methods in S3E as a discipline that helps in a support of the whole information life cycle in organization systems. Despite that, one of the hot issues in practice is still how to effectively develop and maintain complex systems from various aspects, particularly when software components are crucial for addressing declared system goals, and their successful operation. It seems that nowadays we have great theoretical potentials for application of new and more effective approaches in S3E. However, it is more likely that real deployment of such approaches in industry practice is far behind their theoretical potentials.

The main goal of Track 5 is to address open questions and real potentials for various applications of modern approaches and technologies in S3E so as to develop and implement effective software services in a support of information management and system engineering. We intend to address interdisciplinary character of a set of theories, methodologies, processes, architectures, and technologies in disciplines such as: Software Engineering Methods, Techniques, and Technologies, Cyber-Physical Systems, Lean and Agile Software Development, Design of Multimedia and Interaction Systems, Model Driven Approaches in System Development, Development of Effective Software Services and Intelligent Systems, as well as applications in various problem domains. We invite researchers from all over the world who will present their contributions, interdisciplinary approaches or case studies related to modern approaches in S3E. We express an interest in gathering scientists and practitioners interested in applying these disciplines in industry sector, as well as public and government sectors, such as healthcare,

education, or security services. Experts from all sectors are welcomed.

TOPICS

Submissions to S3E are expected from, but not limited to the following topics:

- Advanced methodology approaches in S3E – new research and development issues
- Advanced S3E Process Models
- Applications of S3E in various problem domains – problems and lessons learned
- Applications of S3E in Lean Production and Lean Software Development
- Total Quality Management and Standardization for S3E
- Artificial Intelligence and Machine Learning methods in advancing S3E approaches
- S3E for Information and Business Intelligence Systems
- S3E for Embedded, Agent, Intelligent, Autonomous, and Cyber-Physical Systems
- S3E for Design of Multimedia and Interaction Systems
- S3E with User Experience and Interaction Design Methods
- S3E with Big Data and Data Science methods
- S3E with Blockchain and IoT Systems
- S3E for Cloud and Service-Oriented Systems
- S3E for Smart Data, Smart Products, and Smart Services World
- S3E in Digital Transformation
- Cyber-Physical Systems (9th Workshop IWCPS-9)
- Model Driven Approaches in System Development (7th Workshop MDASD'22)
- Software Engineering (42th IEEE Workshop SEW-42)

TRACK CHAIRS

- **Luković, Ivan**, Unniversity of Belgrade, Serbia
- **Kardas, ,** Geylani, Ege University International Computer Institute, Turkey

PROGRAM CHAIRS

- **Bowen, Jonathan**, Museophile Ltd., United Kingdom
- **Hinchey, Mike**(Lead Chair), Lero-the Irish Software Engineering Research Centre, Ireland
- **Szmuc, Tomasz**, AGH University of Science and Technology, Poland
- **Zalewski, Janusz**, Florida Gulf Coast University, United States
- **Seyed Hossein Haeri, ,** IOG and University of Bergen, Norway

PROGRAM COMMITTEE

- **Ahmad, Muhammad Ovais**, Karlstad University, Sweden
- **Challenger, Moharram**, University of Antwerp, Belgium
- **Dejanović, Igor**, Faculty of Technical Sciences, University of Novi Sad, Serbia
- **Derezinska, Anna**, Institute of Computer Science, Warsaw University of Technology, Poland
- **Dutta, Arpita**, National University of Singapore
- **Erata, Ferhat**, Yale University, USA
- **Escalona, M.J.**, University of Seville, Spain
- **Essebaa, Imane**, Faculté des Sciences et Techniques Mohammedia, Morocco
- **García-Mireles, Gabriel**, Universidad de Sonora, Mexico
- **Göknil, Arda**, SINTEF Digital, Norway
- **Hanslo, Ridwaan**, University of Pretoria, South Africa
- **Jarzewicz, Aleksander**, Gdansk University of Technology, Poland
- **Kaloyanova, Kalinka**, University of Sofia, Bulgaria
- **Katic, Marija**, University of London, United Kingdom
- **Khelif, Wiem**, FSEGS, Tunisia
- **Kolukisa Tarhan, Ayça**, Hacettepe University, Turkey
- **Krdzavac, Nenad**, University of Belgrade, Serbia
- **Marcinkowski, Bartosz**, University of Gdansk, Poland
- **Milašinović, Boris**, Faculty of Electrical Engineering, University of Zagreb, Croatia
- **Milosavljevic, Gordana**, Faculty of Technical Sciences, University of Novi Sad, Serbia
- **Misra, Sanjay**, Ostfold University, Norway
- **Morales Trujillo, Miguel Ehécatl**, University of Canterbury, New Zealand
- **Ozkan, Necmettin**, Kuveyt Turk Participation Bank
- **Ozkaya, Mert**, Istanbul Kemerburgaz University, Turkey
- **Ristic, Sonja**, Faculty of Technical Sciences, University of Novi Sad, Serbia
- **Rossi, Bruno**, Masaryk University, Czech Republic
- **Sanden, Bo**, Colorado Technical University, USA
- **Sierra, Jose Luis**, Universidad Complutense de Madrid, Spain
- **Torrecilla-Salinas, Carlos**, IWT2
- **Varanda Pereira, Maria João**, Instituto Politécnico de Bragança, Portugal
- **Vescoukis, Vassilios**, National Technical University of Athens, Greece

Advances in Software, System and Service Engineering

THE S3E track emphasizes the issues relevant to developing and maintaining software systems that behave reliably, efficiently and effectively. This track investigates both established traditional approaches and modern emerging approaches to large software production and evolution.

For decades, it is still an open question in software industry, how to provide fast and effective software process and software services, and how to come to the software systems, embedded systems, autonomous systems, or cyber-physical systems that will address the open issue of supporting information management process in many, particularly complex organization systems. Even more, it is a hot issue how to provide a synergy between systems in common and software services as mandatory component of each modern organization, particularly in terms of IoT, Big Data, and Industry 4.0 paradigms.

In recent years, we are the witnesses of great movements in the area of software, system and service engineering (S3E). Such movements are both of technological and methodological nature. By this, today we have a huge selection of various technologies, tools, and methods in S3E as a discipline that helps in a support of the whole information life cycle in organization systems. Despite that, one of the hot issues in practice is still how to effectively develop and maintain complex systems from various aspects, particularly when software components are crucial for addressing declared system goals, and their successful operation. It seems that nowadays we have great theoretical potentials for application of new and more effective approaches in S3E. However, it is more likely that real deployment of such approaches in industry practice is far behind their theoretical potentials.

The main goal of Track 5 is to address open questions and real potentials for various applications of modern approaches and technologies in S3E so as to develop and implement effective software services in a support of information management and system engineering. We intend to address interdisciplinary character of a set of theories, methodologies, processes, architectures, and technologies in disciplines such as: Software Engineering Methods, Techniques, and Technologies, Cyber-Physical Systems, Lean and Agile Software Development, Design of Multimedia and Interaction Systems, Model Driven Approaches in System Development, Development of Effective Software Services and Intelligent Systems, as well as applications in various problem domains. We invite researchers from all over the world who will present their contributions, interdisciplinary approaches or case studies related to modern approaches in S3E. We express an interest in gathering scien-

tists and practitioners interested in applying these disciplines in industry sector, as well as public and government sectors, such as healthcare, education, or security services. Experts from all sectors are welcomed.

TOPICS

Submissions to S3E are expected from, but not limited to the following topics:

- Advanced methodology approaches in S3E – new research and development issues
- Advanced S3E Process Models
- Applications of S3E in various problem domains – problems and lessons learned
- Applications of S3E in Lean Production and Lean Software Development
- Total Quality Management and Standardization for S3E
- Artificial Intelligence and Machine Learning methods in advancing S3E approaches
- S3E for Information and Business Intelligence Systems
- S3E for Embedded, Agent, Intelligent, Autonomous, and Cyber-Physical Systems
- S3E for Design of Multimedia and Interaction Systems
- S3E with User Experience and Interaction Design Methods
- S3E with Big Data and Data Science methods
- S3E with Blockchain and IoT Systems
- S3E for Cloud and Service-Oriented Systems
- S3E for Smart Data, Smart Products, and Smart Services World
- S3E in Digital Transformation
- Cyber-Physical Systems (9th Workshop IWCPs-9)
- Model Driven Approaches in System Development (7th Workshop MDASD'22)
- Software Engineering (42th IEEE Workshop SEW-42)

TRACK CHAIRS

- **Luković, Ivan**, Unniversity of Belgrade, Serbia
- **Kardas, ,** Geylani, Ege University International Computer Institute, Turkey

PROGRAM CHAIRS

- **Bowen, Jonathan**, Museophile Ltd., United Kingdom
- **Hinchey, Mike**(Lead Chair), Lero-the Irish Software Engineering Research Centre, Ireland
- **Szmuc, Tomasz**, AGH University of Science and Technology, Poland

- **Zalewski, Janusz**, Florida Gulf Coast University, United States
- **Seyed Hossein Haeri**, IOG and University of Bergen, Norway

PROGRAM COMMITTEE

- **Ahmad, Muhammad Ovais**, Karlstad University, Sweden
- **Challenger, Moharram**, University of Antwerp, Belgium
- **Dejanović, Igor**, Faculty of Technical Sciences, University of Novi Sad, Serbia
- **Derezinska, Anna**, Institute of Computer Science, Warsaw University of Technology, Poland
- **Dutta, Arpita**, National University of Singapore
- **Erata, Ferhat**, Yale University, USA
- **Escalona, M.J.**, University of Seville, Spain
- **Essebaa, Imane**, Faculté des Sciences et Techniques Mohammedia, Morocco
- **García-Mireles, Gabriel**, Universidad de Sonora, Mexico
- **Göknil, Arda**, SINTEF Digital, Norway
- **Hanslo, Ridewaan**, University of Pretoria, South Africa
- **Jarzewicz, Aleksander**, Gdansk University of Technology, Poland
- **Kaloyanova, Kalinka**, University of Sofia, Bulgaria
- **Katic, Marija**, University of London, United Kingdom
- **Khelif, Wiem**, FSEGS, Tunisia
- **Kolukisa Tarhan, Ayça**, Hacettepe University, Turkey
- **Krdzavac, Nenad**, University of Belgrade, Serbia
- **Marcinkowski, Bartosz**, University of Gdansk, Poland
- **Milašinović, Boris**, Faculty of Electrical Engineering, University of Zagreb, Croatia
- **Milosavljevic, Gordana**, Faculty of Technical Sciences, University of Novi Sad, Serbia
- **Misra, Sanjay**, Ostfold University, Norway
- **Morales Trujillo, Miguel Ehécatl**, University of Canterbury, New Zealand
- **Ozkan, Necmettin**, Kuveyt Turk Participation Bank
- **Ozkaya, Mert**, Istanbul Kemerburgaz University, Turkey
- **Ristic, Sonja**, Faculty of Technical Sciences, University of Novi Sad, Serbia
- **Rossi, Bruno**, Masaryk University, Czech Republic
- **Sanden, Bo**, Colorado Technical University, USA
- **Sierra, Jose Luis**, Universidad Complutense de Madrid, Spain
- **Torrecilla-Salinas, Carlos**, IWT2
- **Varanda Pereira, Maria João**, Instituto Politécnico de Bragança, Portugal
- **Vescoukis, Vassilios**, National Technical University of Athens, Greece

Extensible Conflict-Free Replicated Datatypes for Real-time Collaborative Software Engineering

Istvan David, Eugene Syriani, Constantin Masson

Department of Computer Science and Operations Research (DIRO) – Université de Montréal, Canada

Email: istvan.david@umontreal.ca, syriani@iro.umontreal.ca

Abstract—Real-time collaboration has become a prominent feature of nowadays’ software engineering practices. Conflict-free replicated data types (CRDT) offer efficient mechanisms for implementing real-time collaborative environments. However, the lack of extensibility of CRDT limits their applicability. This is a particularly important problem in settings relying on complex, non-linear data types. In this paper, we report our results in augmenting primitive CRDT with extension mechanisms. We demonstrate our technique through an example from the realm of model-driven engineering, where graph types are prevalent.

Index Terms—CRDT, Collaborative software engineering, Strong eventual consistency, Concurrency control

I. INTRODUCTION

TODAY’S software engineering is often carried out in distributed settings [1], necessitating advanced coordination mechanisms, such as real-time collaboration. A key challenge in real-time collaboration is to guarantee the convergence of the stakeholders’ local replicas while ensuring timely execution and smooth user experience [2]. Traditional mechanisms that implement locking [3] fall short of addressing this challenge. A more appropriate consistency model for real-time collaboration is strong eventual consistency (SEC) [4]. SEC ensures that (i) the updates will eventually be observed by each stakeholder, and (ii) the local replicas that received the same updates will be in the same state.

Conflict-free replicated Data Types (CRDT) [5] provide an efficient implementation of the SEC model. While CRDT have been successful in supporting real-time collaboration over linear data types, such as text and source code, some applications require more complex data types. For example, graph models are frequently employed in Model-Driven Software Engineering (MDSE) [6]. However, the lack of appropriate extension mechanisms in current CRDT frameworks renders the definition of more complex data types a challenging task.

In this paper, we report on our experiments with extensible CRDT using our prototype framework, CollabServer¹. The contributions include (i) a collection of CRDT primitives; (ii) an extension mechanism for the customization of CRDT; and (iii) performance assessment of the approach. We demonstrate our approach on an illustrative case for Mind map editors, which represents typical modeling environments that require graph semantics to represent models.

C. Masson is currently with Ubisoft, Paris.

¹<https://github.com/geodes-sms/collabserver-framework>

II. BACKGROUND

Collaborative Model-Driven Software Engineering: The challenges of distributed software engineering are substantially exacerbated by the complexity of the engineered system that necessitates collaboration between stakeholders of highly diverse expertise. Model-driven software engineering (MDSE) [6] allows stakeholders to reason at higher levels of abstraction and by that, enables aligning the work of diverse stakeholders. Combining the benefits of collaborative software engineering with MDSE, collaborative MDSE [7] has become a prominent paradigm in software engineering practice. Due to the often disparate vocabularies of stakeholders, identifying overlaps between the stakeholders’ concerns is not trivial [8]. This, in turn, renders the detection of conflicts a challenging task. Recent studies [9], [10] show that only one-third of real-time collaborative MDSE solutions provide explicit conflict resolution mechanisms. State-of-the-art tools mostly rely on version control systems to facilitate collaborative MDSE [11]. Other approaches define consistency in terms of bijective correspondence, e.g., by linking elements through correspondence graphs [12], processes [13], or semantic links [14]. However, these techniques do not support real-time collaboration.

Real-time collaboration and CRDT: Sun et al. [2] define four requirements for effective real-time collaboration: (i) convergence of replicas; (ii) user intention preservation; (iii) causality preservation of updates; and (iv) timely execution of updates. CRDT support real-time collaboration by eliminating conflicts between the distributed stakeholders’ operations, without the need for a costly consensus mechanism, while showcasing excellent fault tolerance and reliability properties. Notable open-source CRDT frameworks include Automerge² and Yjs³. Automerge enables real-time collaboration in JavaScript-based systems, based on the JSON format. Yjs uses linked lists as its foundational data type, but the internal representation can be extended to achieve collaboration over complex data types. However, this extension is not trivial.

III. THE COLLABSERVER FRAMEWORK

A. Design principles

1) *Operation-based CRDT*: There are two equivalent approaches to implementing CRDT. State-based CRDT are

²<https://github.com/automerge/automerge>

³<https://github.com/yjs/yjs>

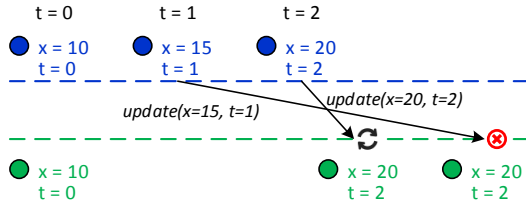


Fig. 1. Total order of updates in the LWW paradigm.

structured in a way that they adhere to a monotonic semi-lattice. Operation-based CRDT require that concurrent operations commute, i.e., irrespective of the order of operations, local replicas converge. We have opted for the operation-based CRDT scheme. This approach, as opposed to state-based CRDT, requires less network bandwidth, because only operations have to be sent through the network. In addition, an operation-based approach is more suitable for integration with external components, such as databases and user interfaces.

2) *Last-Writer-Wins (LWW)*: Automated reconciliation between replicas can be achieved at the application level or at the data level [15]. The LWW paradigm [16] implements the latter and has been widely adopted for operation-based conflict resolution. In LWW, conflicted operations are resolved by timestamps; the data with the more recent timestamp prevails. Fig. 1 shows an example resolution scenario under LWW. User A and User B initially have their local replicas in consistent states. At $t = 1$, User A executes the update $x = 15$. The updated value and the timestamp are propagated to User B. However, before the message arrives, User A executes another update: $x = 20$, at time $t = 2$. The update is propagated to User B. Networks usually do not guarantee order preservation. Thus, the second update may arrive at User B earlier than the first. Upon receiving the update, User B will reconcile this new value with his local replica. User B has $x = 10$ timestamped with $t = 0$; and an update that says $x = 20$ timestamped with $t = 2$. Under the LWW paradigm, the latter value is accepted. Eventually, the first update reaches User B. User B has $x = 20$ timestamped with $t = 2$; and an update of $x = 15$ timestamped with $t = 1$. Under the LWW paradigm, the former value is accepted, leaving the replicas in consistent states.

3) *CRDT tombstone metadata*: To ensure that operations commute, data is never deleted, only flagged as removed (i.e., soft delete). This information is captured in the tombstone metadata with boolean semantics. In an alternative approach by Shapiro et al. [5], dedicated partitions of the specific datatypes are reserved for storing deleted elements (LWW-element-Set). The benefit of our approach is that it reduces the number of elementary data operations upon changes.

4) *Commutativity and idempotence*: Operation-based CRDT assume that operations commute (i.e., $x \circ y = y \circ x$) and are idempotent (i.e., $x \circ x = x$). Traditionally, these properties are ensured by the communication protocol [5]. We have implemented the base type system of CollabServer in a way that commutativity and idempotence are guaranteed by design. As a consequence, CRDT that extend base

TABLE I
COLLABSERVER BASIC TYPES AND THEIR METHODS

Data type	Methods
<i>LWWRegister</i>	<code>query()</code> <code>update(value, timestamp)</code>
<i>LWWSet</i>	<code>query(key)</code> <code>add(key, timestamp)</code> <code>clear(timestamp)</code> <code>remove(key, timestamp)</code>
<i>LWWMap</i>	<code>query(key)</code> <code>add(key, value, timestamp)</code> <code>clear(timestamp)</code> <code>remove(key, timestamp)</code>
<i>LWWGraph</i>	<code>queryVertex(vertexID)</code> <code>addVertex(vertexID, timestamp)</code> <code>removeVertex(vertexID, timestamp)</code> <code>addEdge(source, target, timestamp)</code> <code>removeEdge(source, target, timestamp)</code> <code>clearVertices(timestamp)</code>

CollabServer types are expected to satisfy these properties without further development effort. Commutativity and idempotence are achieved by the combination of *timestamps* and *tombstones*. Timestamps ensure that each replica will order the update operations in the same way. Tombstone metadata ensures that no information is lost.

B. CollabServer CRDT primitives

Table I summarizes the CRDT provided by CollabServer. Every CRDT is equipped with atomic create, read, update, and delete (CRUD) operations. More complex operations can be implemented in specific applications. CollabServer is implemented in C++, using the Standard Template Library (STL) [17]. At the source code level, CollabServer CRDT are implemented as C++ templates, allowing easy extensibility and customization. More information is available in [18] and from the GitHub repository of the project¹. In the following, we briefly discuss the CRDT provided by CollabServer.

1) *LWWRegister*: The *LWWRegister* is the simplest primitive implemented in the CollabServer framework. It stores an atomic value, its timestamp ts , and its tombstone metadata. The `query` method returns the value stored in the register. The `update` method changes this value, as shown in Algorithm 1.

2) *LWWSet*: The *LWWSet* is a monotonically increasing data structure with the usual set semantics. That is, a value can exist in the set only once. The *LWWSet* is implemented as a C++ `HashMap`, with the values stored in the key set, and the associated value set storing the metadata (timestamp and tombstone). The `query` method (Algorithm 2) returns the respective key of the hashmap if it exists and is not marked as removed. The `add` method (Algorithm 3) inserts an element into the set. If the element already exists in the set, its timestamp is updated. If the element does not exist in the set, it is added to the set, along with the required metadata. The `remove` method (Algorithm 4) flags an element as deleted if its `timestamp` is higher than the current timestamp. In case the element is already deleted, its timestamp is updated, and a `false` value is returned. If the requested element is not present in the set, it is added with tombstone metadata that designates a deleted state. The `clear` method executes the `remove` method on every element in the set.

Algorithm 1: `lwregister_update(value, ts)`

```

if value, ts > current_timestamp then
  current_value = value
  current_timestamp = value, ts
  return true
else
  return false

```

Algorithm 2: `lwset_query(key)`

```

element = hashmap[key]
if element is not None AND is not element.value.isRemoved then
  return element.key
else
  return None

```

Algorithm 3: `lwset_add(key, ts)`

```

element = hashmap[key]
if element is not None then
  if ts > element.value.timestamp then
    element.value.timestamp = ts
    if element.value.isRemoved then
      element.value.isRemoved = false
    return true
  return false
else
  element = {key, {ts, false}}
  hashmap.add(element)
  if element.value.timestamp <= lastclear_timestamp then
    element.value.timestamp = lastclear_timestamp
    element.value.isRemoved = true
    return false
  else
    return true

```

3) *LWWMap*: The *LWWMap* stores key-value pairs of data with the keys being stored in an *LWWSet* and the associated value being stored in an *LWWRegister*. By reusing the previously defined LWW types, the API methods of the *LWWMap* can be reduced to the ones defined in Algorithms 1–4.

IV. CUSTOMIZING CRDT

In this section, we demonstrate the extensibility of CollabServer CRDT by (i) constructing a custom physical CRDT, the *LWWGraph* (Section IV-A); and (ii) based on this custom type, constructing a domain-specific type (Section IV-B). For the latter, we use the example of a Mind map editor, providing domain-specific operations for constructing and manipulating a Mind map, such as adding and removing topics; and placing a marker on a topic. Fig. 2 show the extended type system.

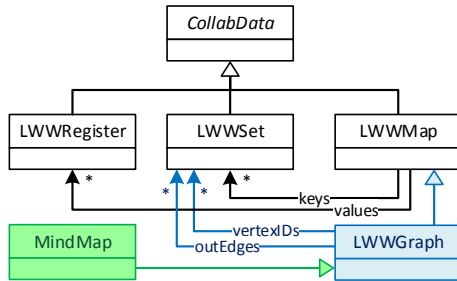


Fig. 2. Type system with the customized types highlighted

A. Constructing custom CRDT: *LWWGraph*

The *LWWGraph* is a directed graph, represented by its adjacency list, stored in an *LWWMap*. Vertex IDs are stored as keys and the vertices are stored as values. There are no restrictions on the type of the vertex ID, it only depends on the specific implementation, and its typing is deferred to the developer. A vertex is defined as a tuple (*content, edges*), describing the content of the vertex and an *LWWSet* of the outgoing edges from this vertex. Each key is the ID of the target vertex. The `queryVertex` method invokes the `query` method on the *LWWMap* storing the adjacency list of vertices, and returns the vertex if exists. Similarly, the `addVertex` method invokes the

Algorithm 4: `lwset_remove(key, ts)`

```

element = hashmap[key]
if element is not None then
  if ts > element.value.timestamp then
    element.value.timestamp = ts
    if not element.value.isRemoved then
      element.value.isRemoved = true
      return true
  return false
else
  element = {key, {ts, true}}
  hashmap.add(element)
  return false

```

`add` method on the *LWWMap* storing the adjacency list. The `removeVertex` method (Algorithm 5) removes a vertex from the graph, and all edges connected to it. If the vertex does not exist yet, it is added in the adjacency list, and the `remove` method of the *LWWMap* is invoked. The `addEdge` method (Algorithm 6) creates a new edge connecting the source vertex with the target vertex. We distinguish between three scenarios. First, we apply `addEdge` on two existing vertices. In this case, the edge is added to the underlying graph; or if the edge already exists, its timestamp is updated. Second, we apply `addEdge` when one or two vertices do not exist. This case occurs when the `addEdge` operation is observed before the `addVertex` operation. To ensure the commutativity of operations, CollabServer implements the `addEdge` method in a way that it also applies `addVertex` on the source and target vertices. Missing vertices are then simply added along with the edge. Receiving a later `addVertex` operation will simply update the timestamps. Third, we apply `addEdge` with the source and/or the target vertex that has already been deleted. This case occurs when the `addEdge` operation is received with the source and/or the target vertex already deleted. The case where `removeVertex` is older than `addEdge` is a trivial one, since `addEdge` also applies `addVertex` as seen earlier. The opposite case (`removeVertex` older than `addEdge`) requires additional steps. First, the edge is created as discussed before.

Algorithm 5: `lwwgraph_removeVertex(vertexID,ts)`

```

removed = adj.remove(vertexID, ts)
vertex = adj.queryCRDT(vertexID)
vertex.edges.clear(ts)
for vertex in adj.iteratorCRDT do
  if vertex.edges.has(vertexID) then
    vertex.edges.remove(vertexID, ts)
return removed

```

Algorithm 6: `lwwgraph_addEdge(src, trgt, ts)`

```

info = {'srcAdded': false, 'trgtAdded': false, 'edgeAdded': false}
info['srcAdded'] = adj.add(src, ts)
info['trgtAdded'] = adj.add(trgt, ts)
vertexSrc = adj.queryCRDT(src)
info['edgeAdded'] = vertexSrc.edges.add(trgt, ts)
if not vertexSrc.edges.queryCRDT(trgt).isRemoved then
  vertexTrgt = adj.queryCRDT(trgt)
  if vertexSrc.isRemoved OR vertexTrgt.isRemoved then
    t = max(vertexSrc.ts, vertexTrgt.ts)
    vertexSrc.edges.remove(trgt, t)
    info.edgeAdded = false
  return info
return info

```

Then, we check if the newly created edge is dangling and remove it. This way, `addEdge` is commutative. Note that, as shown in Algorithm 6, this operation returns more information since additional actions can be performed on the edge or the vertices. The `removeEdge` method (Algorithm 7) removes an edge from the graph. This operation may encounter a situation where the source vertex does not exist yet in the graph. Since all operations are required to be commutative, the source vertex is created with the smallest timestamp and the `isRemoved` flag is set to true. The `clearVertices` method removes all vertices and their associated edges from the graph.

B. Constructing domain-specific CRDT

The construction of custom CRDT is achieved by extending the `CollabData` base type and defining custom operations while ensuring their commutative and idempotent properties. We demonstrate the extensibility of primitive `CollabServer` data types by constructing the *MindMap* CRDT for the `MindmapEditor` application. The *MindMap* type is a graph; each topic and marker of the *MindMap* is a vertex of the graph; edges of the graph connect topics to their parent topic, and markers to the topics they mark. Constructing the *MindMap* requires extending the *LWWGraph* primitive type and augmenting it with domain-specific methods that are commutative and idempotent. The outline of the *MindMap* CRDT is shown in Listing 1. The full implementation is available from the GitHub repository of the project.¹

The methods in Listing 1 reuse the API of the *LWWGraph*; thus, they inherit CRDT properties. The *MindMap* type extends the *LWWGraph* by introducing the notion of *attributes*, for example, for storing the *name* of the *MindMap*. As shown

Algorithm 7: `lwwgraph_removeEdge(src, trgt, ts)`

```

adj.remove(src, Timestamp.MIN)
if src != trgt then
  adj.remove(trgt, Timestamp.MIN)
vertex = adj.queryCRDT(src)
return vertex.edges.remove(trgt, ts)

```

in Listing 2, attributes are added to *LWWGraph*-derivatives by invoking the `add` method of the *LWWMap* that allows storing key-value pairs. The built-in CRDT can be readily reused to construct custom data types. This has been demonstrated in this example, and also in the definition of the *LWWMap* and *LWWGraph* that reuse more primitive `CollabServer` CRDT.

Listing 1. *MindMap* CRDT with domain-specific operations

```

class Topic: Vertex{...}
class Marker: Vertex{...}

class MindMap: LWWGraph{
  void addTopic(const UUID& topicId){
    LWWGraph::addVertex(topicId, Timestamp::now())
    notifyOperationBroadcaster()
  }
  void removeTopic(const UUID& id){}
  void addMarker(const UUID& id){}
  void removeMarker(const UUID& id){}
  void connectTopics(const UUID& t1, const UUID& t2){
    LWWGraph::addEdge(t1, t2, Timestamp::now())
    notifyOperationBroadcaster()
  }
  void putMarker(const UUID& m, const UUID& t){}
  ...
}

```

Listing 2. API for adding attributes to the *MindMap*

```

class MindMap: LWWGraph{
  void addAttribute(
    const UUID& id,
    const std::string& name,
    const std::string& value) {
    //calls the LWWMap super class
    LWWGraph::add(name, value, Timestamp::now())
    notifyOperationBroadcaster()
  }
}

```

V. PERFORMANCE EVALUATION

Although performance was not our primary concern in this exploratory project, we provide a performance evaluation of the framework. As the performance of CRDT is determined by the number of objects present in the application [19], we assess the performance by simulating a scenario in which new vertices and edges are added to a shared model.

Experimental setup: We used the following sequence as a test scenario: $\text{add topic}_i \rightarrow \text{add topic}_j \rightarrow \text{connectTopics topic}_i, \text{topic}_j$. The scenario was executed 50.000 times. That is, 100.000 topics (graph vertices) and 50.000 relationships (edges) were generated. We have executed the test scenario with one, two, and four parallel users, and measured the change in response times. In the case of two and four parallel simulated users, each user carried out

TABLE II
RESPONSE TIMES OF THE 1/2/4 USER CASES

Users	Min [ms]	Response times		
		μ [ms]	σ	Max [ms]
1	0.11	0.94	0.22	17.4
2	0.13	2.06	0.33	17.4
4	0.37	5.92	0.49	17.4

this sequence, adding topics (vertices) and connecting them (by adding edges) to the same shared mind map (graph). The measurements have been executed on a VMWare virtual machine running a 64-bit Ubuntu 20.04.1. OS, with 4GB of memory, and with 4 CPU cores allocated, checked at 2.6 GHz.

Results: To assess the *scalability* of the framework, response times were measured at the local replicas, defined as the time difference between issuing a command in the editor and getting a response. To filter noise, we clipped the sample at $\mu \pm 3\sigma$ (0.2% of the cases). The mean response time in the one-user case shows linear scaling with the number of objects. (Linear regression statistic: $p = 22E - 17$.) We have observed the same linear increase in response times in the two and four-user cases. We have also observed increasing response times with the increasing number of users. Table II shows the mean response time in one, two, and four user cases. The mean response time increased by a factor of 2.2 and 2.8 as the number of users doubled from one to two, and from two to four, respectively. A statistically significant difference is observed in the mean response time of the three cases, as confirmed by a t-test ($\alpha = 0.05$, $p = 2e-11$).

We observed a linear increase in the *memory heap*. The majority of memory consumption is due to C++ node iterators and hashtable objects the `LWWGraph` relies on.

Discussion: We observe a linearly increasing response time and a linearly increasing memory footprint. This is in line with the observation of Sun et al. [20]. We conclude that this performance profile is characteristic of CRDT implementations, and can be effectively treated by suitable garbage collection mechanisms [21]. We consider these results adequate (i) considering the benefits in extensibility `CollabServer` CRDT provide; and (ii) considering that performance was not the primary objective of the current solution.

VI. CONCLUSION

In this paper, we presented an approach for augmenting CRDT with extension mechanisms and demonstrated that the performance repercussions of extensibility are manageable. We provided a family of concurrency control algorithms, ensuring strong eventual consistency and allowing for efficient real-time collaboration. Our algorithms and data types show linear scaling of response time and memory footprint with the number of objects in memory and with users. This is a characteristic performance profile of CRDT. Our results suggest that CRDT can be used in disciplines where customizability is a key factor, such as collaborative modeling using domain-specific modeling languages. In future work, we will investigate garbage collection mechanisms to achieve the

scalability collaborative engineering tools require. We used the takeaways of this exploratory project in the development of our real-time collaborative modeling framework `lowkey` [22].

REFERENCES

- [1] J. Whitehead, "Collaboration in software engineering: A roadmap," in *Future of Software Engineering*. IEEE, 2007. doi: 10.1007/978-3-642-10294-3_1 pp. 214–225.
- [2] C. Sun *et al.*, "Achieving Convergence, Causality Preservation, and Intention Preservation in Real-Time Cooperative Editing Systems," *ACM Trans. Comput.-Hum. Interact.*, vol. 5, no. 1, p. 63–108, 1998.
- [3] P. A. Bernstein and N. Goodman, "Concurrency Control in Distributed Database Systems," *ACM Comput. Surv.*, vol. 13, no. 2, pp. 185–221, 1981. doi: 10.1145/356842.356846
- [4] V. B. Gomes *et al.*, "Verifying strong eventual consistency in distributed systems," *Proceedings of the ACM on Programming Languages*, vol. 1, no. OOPSLA, pp. 1–28, 2017. doi: 10.1145/3133933
- [5] M. Shapiro, N. Preguica, C. Baquero, and M. Zawirski, "Conflict-free replicated data types," in *Symposium on Self-Stabilizing Systems*. Springer, 2011. doi: 10.1007/978-3-642-24550-3_29 pp. 386–400.
- [6] D. C. Schmidt, "Model-Driven Engineering," *IEEE Computer*, vol. 39, no. 2, pp. 25–31, 2006. doi: 10.1109/MC.2006.58
- [7] H. Muccini, J. Bosch, and A. van der Hoek, "Collaborative modeling in software engineering," *IEEE Software*, vol. 35, no. 6, pp. 20–24, 2018.
- [8] I. David *et al.*, "Engineering process transformation to manage (in)consistency," in *Proceedings of the 1st Intl. Workshop on Collaborative Modelling in MDE*, vol. 1717. CEUR-WS.org, 2016, pp. 7–16.
- [9] —, "Collaborative Model-Driven Software Engineering: A Systematic Update," in *Proceedings of the 24th ACM/IEEE International Conference on Model Driven Engineering Languages and Systems*. ACM, 2021. doi: 10.1109/MODELS50736.2021.00035 pp. 273–284.
- [10] M. Franzago, D. D. Ruscio, I. Malavolta, and H. Muccini, "Collaborative model-driven software engineering: A classification framework and a research map," vol. 44, no. 12, 2018. doi: 10.1109/TSE.2017.2755039 pp. 1146–1175.
- [11] S. Kelly, "Collaborative modelling with version control," in *Federation of Intl. Conferences on Software Technologies: Applications and Foundations*. Springer, 2017. doi: 10.1007/978-3-319-74730-9_3 pp. 20–29.
- [12] C. Adourian and H. Vangheluwe, "Consistency between geometric and dynamic views of a mechanical system," in *Proceedings of the 2007 Summer Computer Simulation Conference*. Society for Computer Simulation International, 2007.
- [13] I. David *et al.*, "Towards inconsistency management by process-oriented dependency modeling," in *Proceedings of the 9th International Workshop on Multi-Paradigm Modeling, 2015*, ser. CEUR Workshop Proceedings, vol. 1511. CEUR-WS.org, 2015, pp. 32–41.
- [14] K. Vanherpen *et al.*, "Ontological reasoning for consistency in the design of cyber-physical systems," in *1st International Workshop on Cyber-Physical Production Systems*. IEEE, 2016, pp. 1–8.
- [15] C. Meiklejohn and P. Van Roy, "Lasp: A language for distributed, coordination-free programming," in *Proceedings of the 17th International Symposium on Principles and Practice of Declarative Programming*. ACM, 2015. doi: 10.1145/2790449.2790525 p. 184–195.
- [16] P. R. Johnson and R. Thomas, *RFC0677: Maintenance of duplicate databases*. RFC Editor, 1975.
- [17] A. Stepanov and M. Lee, *The standard template library*. HP Laboratories 1501 Page Mill Road, Palo Alto, CA 94304, 1995, vol. 1501.
- [18] C. Masson, "Framework for Real-time collaboration on extensive Data Types using Strong Eventual Consistency," Master's thesis, Université de Montréal, Canada, December 2018.
- [19] D. Sun and C. Sun, "Operation Context and Context-based Operational Transformation," in *Proceedings of the 2006 20th Conference on Computer Supported Cooperative Work*. ACM, 2006. doi: 10.1145/1180875.1180918 pp. 279–288.
- [20] D. Sun, C. Sun, A. Ng, and W. Cai, "Real Differences between OT and CRDT in Correctness and Complexity for Consistency Maintenance in Co-Editors," *Proc. ACM Hum.-Comput. Interact.*, vol. 4, 2020.
- [21] J. Bauwens and E. Gonzalez Boix, "Memory Efficient CRDTs in Dynamic Environments," in *Proceedings of the 11th ACM SIGPLAN International Workshop on Virtual Machines and Intermediate Languages*, ser. VMIL 2019. ACM, 2019. doi: 10.1145/3358504.3361231 p. 48–57.
- [22] I. David and E. Syriani, "Real-time Collaborative Multi-Level Modeling by Conflict-Free Replicated Data Types," *Softw. Syst. Model.*, 2022.

Sensor Data Protection in Cyber-Physical Systems

1st Anton Hristozov
 Polytechnic Institute
 Purdue University

West Lafayette, Indiana, USA
 ahristoz@purdue.edu

2nd Dr. Eric Matson
 Polytechnic Institute
 Purdue University

West Lafayette, Indiana, USA
 ematson@purdue.edu

3rd Dr. Eric Dietz
 Polytechnic Institute
 Purdue University

West Lafayette, Indiana, USA
 jedietz@purdue.edu

4th Dr. Marcus Rogers
 Polytechnic Institute
 Purdue University

West Lafayette, Indiana, USA
 rogersmk@purdue.edu

Abstract—Cyber-Physical Systems (CPS) have a physical part that can interact with sensors and actuators. The data that is read from sensors and the one generated to drive actuators is crucial for the correct operation of this class of devices. Most implementations trust the data being read from sensors and the outputted data to actuators. Real-time validation of the input and output of data for any system is crucial for the safety of its operation. This paper proposes an architecture for handling this issue through smart data guards detached from sensors and controllers and acting solely on the data. This mitigates potential issues of malfunctioning sensors and intentional sensor and controller attacks. The data guards understand the expected data, can detect anomalies and can correct them in real-time. This approach adds more guarantees for fault-tolerant behavior in the presence of attacks and sensor failures.

Index Terms—CPS, robots, software architecture, fault tolerance, resilience, ROS

I. INTRODUCTION

IT IS not always possible to trust sensor data because of reliability issues in sensors, intentional sensor attacks, and issues like EMI interference [1]. Since sensor data are used in control loops, it is better if we could make sure that the data is not compromised and has not deviated from an acceptable range. This problem is very pertinent for control algorithms. In the era of AI solutions, the data streams can determine the success or failure of neural networks or other machine learning algorithms used in the device. Therefore the issue applies to ML solutions.

Using data guards exploits a separation of concerns approach, applicable to off-the-shelf controllers or AI solutions that are complex and hard to understand. Data guards are much more straightforward in their operation and focus on guarding the validity of the data, independently of how complex the controllers or sensors are. Tuning such complex components is difficult and guaranteeing that they will work in all conditions is sometimes impossible. Therefore using data guards before data is fed to the controller is a security and reliability guarantee which enables safer system operation. A further benefit of separating data guards from the rest of the system is that they can be validated separately, as they tend not to contain too much code and complexity. In this respect, they can be similar to the enforcers presented in the literature [2] but are focused on data instead of behavior.

The paper’s main contribution is the proposal of dedicated smart data guards that take care of sensor data in real-time.

The definition of data contracts is another contribution. The enforcement of such contracts in an existing architecture is a way to enhance systems and enforce security and safety properties. The contribution related to this is that we propose to use separation of concerns through data-centric components that abide by the data contracts we define, depending on the data.

The paper starts with analyzing what data protection means and why we need it. Then the next section discusses data contracts and provides formal presentation examples. The following section discusses how they can improve an existing control architecture. A data guard implementation is also discussed. A reference implementation in PX4 is next. The paper ends with future directions and a conclusion section.

II. DATA PROTECTION ANALYSIS

Sensor data requires attention since it is used for control decisions and can affect the safety of a CPS and is therefore critical. Controllers and observers operate by using sensor data, trusting it is correct. Most of the controllers are designed with the assumption of data correctness. In reality, possible attacks and noise in the data, as well as sensor degradation and failure, are facts that cannot be ignored. Both sources of sensor data incorrectness can be dealt with if we take precautions to validate the data in real-time.

A. Sensor Attacks

If we consider a Cyber-Physical System such as an autonomous vehicle or a UAV, there are generally many sensors used by such a system. Some sensors, such as cameras and other object detection sensors, may need to be processed by complex machine learning algorithms to integrate them into the system. Many other sensors, though, are simpler and can generate fewer data per unit of time. The main issue with trusting sensor data is that there is no way to know if the data are valid since there is usually no authentication and encryption of the data sent from the sensors to the controller. There is also no guarantee that the measured physical value is not affected by an attack. A class of attacks can affect physical values without coming into contact with the sensors using EMI or acoustic waves, for example, [3].

An example of a possible attack is an EMI burst that disrupts a sensor ([4]). The other likely attack can happen when the measured data are transferred to the controller, for example,

through CAN bus [5]. Flipping bits or controlling the sensor data bus can be even an easier way to perform an attack. Sensor buses can be in many different forms and can be, for example, i2c or Spi or a dedicated Ethernet and even a wireless connection in some cases [6]. Given that the attack surface is large, we cannot and should not trust sensor data for safety-critical systems. There is a need to add a mechanism to detect and remedy the effects of an attack that manipulates sensor data.

The data coming from sensors could also be encrypted as a security measure, but this is a pretty resource-intensive operation and can severely affect the timing of transporting the data and using it [7]. Controllers are sensitive to delays, and this approach may become impractical for resource-constrained CPS [8]. There are also hardware solutions that enable encryption, but it is unlikely that all the sensors in an AV can be equipped with such capabilities. Therefore, for this study, we can assume that sensor data travels in the open and can be vulnerable to attacks.

B. Data Guard Components

A sensor typically sends a digital stream of bytes representing a physical parameter, for example, position, velocity, or acceleration in the case of UAVs. In the event of a sensor attack or sensor malfunctioning, we can have data that is not physically realistic at a particular moment, based on the system's state. A data guard can use sensor-specific parameters to guarantee the sensor data [9]. For example, the following parameters can be used in a simple universal approach to sensor validation:

- MAX value - The maximum allowed value for all cases
- MIN value - The minimum allowed value for all cases
- MAX delta - The maximum change in unit time
- MIN delta - The minimum change in unit time
- MAX time for stale data - The maximum time when data can stay the same.
- DEFAULT safe value - When the input value is out of bounds or stale, this value can be fed to the controller. Note that in addition to static default values, we can calculate a default based on historical data when we consider that the system operates under normal conditions.

C. Sensor Data

As one possible example we can have a look at a GPS message in the PX4 autopilot which has the following message abbreviated format in Listing 1:

Listing 1 GPS message details

```
uint64 timestamp
int32 lat
int32 lon
int32 alt
...
```

The individual fields of the GPS message are either integers or real numbers. They can be validated as each message

arrives in a software component as part of the system. Such an approach takes care of each message instance but does not help check data deviations between successive message instances. We need an algorithm that contains meta parameters used for all messages, such as delta max, delta min, stale timer values, and default values. All these parameters can be considered as data contract parameters. Each data guard expects the sensor to provide data that is within the data contract parameters [10]. Defining and enforcing such contracts is the main contribution of the paper.

D. Data Guard Utilization

Data guard components can work independently from other components in separate threads. The goal is to provide minimal overhead and be transparent to the rest of the control system. The main goal is to have the guards provide reasonable values when the data stream contains unexpected values. In other words, this is not just filtering specific values but actively reconstructing the sensor data when deemed incorrect. This takes care of spikes or short attacks and can even be used to detect a persistent sensor attack. For example, if the data guard keeps a default value for a certain period, it can then generate a signal indicating that something has gone wrong with that sensor. This is a relatively simple implementation but general enough to be used with many different sensors.

III. DATA CONTRACTS

Software contracts have been used in many aspects of software engineering, especially in designing object-oriented systems [11], [12]. In this work, we extend the software contract concepts to be used for the specification of data guards, used to guarantee specific properties in the sensor data and the data sent to actuators. Contracts are based on assumptions and guarantees and can be applied to software interfaces. The general representation is given through equation 1 [11], [12]:

$$C = A, G \quad (1)$$

Where C is the contract, A is the set of assumptions, and G is the set of guarantees. The contract definition can happen during design time since the sensor and actuator data are usually known to the designer. This allows for a thorough analysis of the data and the associated data contract. We start by defining the assumptions A of our data guard contract can be different for different cases and can be expressed as a set according to 2.

$$A = \{A_i\} \quad (2)$$

The data set of each data source by the following set in 3:

$$D = \{D_i\} \quad (3)$$

, where each data member can have a different type.

Some common assumptions for the data in the set D are:

- expected data should have no overflow or underflow for these data types.

- Another assumption can be that new data will be received with a certain minimum frequency.

The guarantees G can include a different set for different data. According to our running example, the guarantees G can be composed through a set of rules 4:

$$G = \{G_i\} \quad (4)$$

An example set of guarantees based on our example follows:

- No data item will exceed its expected maximum value
- No data item will become less than its expected minimum value
- No two successive values will exceed the maximum allowed delta for that data item. The delta is the difference between two consecutive readings.
- A data item will not be stale longer than the maximum allowed number of readings
- Any reading should not deviate from the average of the last N readings by a certain delta value.
- When any of the above rules are violated, a default value will be provided for each data item

Some examples from the set of guarantees G can be represented mathematically in the following way:

$$G_1 : D_i \leq D_{imax},$$

$$G_2 : D_i \geq D_{imin},$$

$$G_3 : D_{i2} - D_{i1} \leq D_{i\delta}$$

$$G_4 : D_i - D_j \neq 0, \text{ where } j = i + k \text{ over } k \text{ time periods}$$

$$G_5 : D_i \in D_u \Rightarrow D_i = D_d$$

where D_u is unacceptable value and D_d is default value

IV. ATTACK RESISTANT CONTROL ARCHITECTURE

Using our defined data contract from the previous section, we can show the proposed control architecture in figure 1. This type of control architecture is typical for a variety of CPS, including UAVs [13]. It is very similar to the classic control loop architecture with the addition of data guard components for each sensor and a data guard for the controller output. Having a data guard for each sensor makes handling each sensor data stream's timing and specific data characteristics easier. This also allows for the sensor fusion to be done separately. The approach assumes that all data guards will be running in parallel so that the streams coming from sensors and the controller can work independently. It is also possible to place data guards after the fusion block; in some situations, this may be a better approach.

Figure 2 shows a possible architecture of a generic data guard. There are two independent timers for calculating the deltas and for the detection of stale data. Stale data means a faulty sensor or a sensor under constant attack. These parameters can be part of the data contract established for each sensor data stream in the system. The data guard component checks for range violations in each message as well as jumps

in data readings between messages that are not realistic, given the characteristics of a particular sensor. For example, an accelerometer cannot generate impossible values given the abilities of a UAV or UGV.

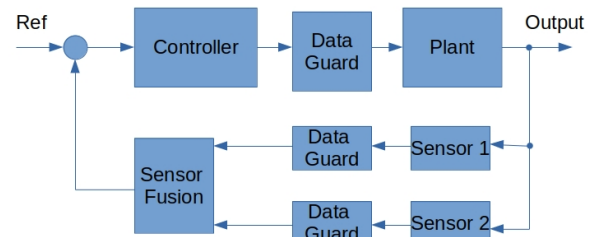


Fig. 1. Control architecture

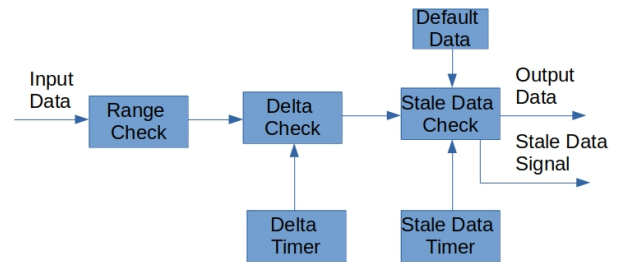


Fig. 2. Data Guard Component

A. Implementation of a Data Guard

Data guards can be implemented in a separate module using a variety of programming languages. For existing systems, they will most likely be in the programming language used to develop the system. Our implementation of a data guard follows the generic approach shown in figure 2. An example algorithm for a data guard is written in pseudo-code in Listing 2. This algorithm utilizes two timers and works continuously on incoming data. The implementation can protect against sudden changes that are not physically expected and can detect if a sensor is faulty and does not function anymore. Generating a signal to the system can be used to make a higher-level decision to enter a different fail-safe mode of control. In that respect, the data guard can be the first level in

Listing 2 Pseudo-code example of a data guard

```

read_value()
if data > max or data < min then
data = default_data
endif
if delta_timer_expiration()
check_delta_max()
check_delta_min()
endif
if delta > delta_max
or delta < delta-min then
data = default_data
endif
if stale_time_expired()
check_for_stale_data()
endif
if stale_data() then
send_stale_alarm()
endif

```

the decision-making when implementing fault-tolerant system-wide behavior.

B. PX4 autopilot as a prototyping platform

PX4 can be used within a Software in the Loop (SITL) environment with Jmavsim, or the Gazebo flight simulator [14]. In either case, the sensors of the drone are part of the simulator, and the data from them is sent periodically to PX4, where the data is analyzed and dispatched to other modules. This is done via the Mavlink protocol, which is a standard protocol for messaging in UAVs [15]. Adding a new custom module and intercepting the data stream from one or more sensors is relatively easy, and this is the chosen approach for the experiments. Similarly, defining new messages and writing code for them is very well supported, and we took advantage of it in this work.

V. REFERENCE IMPLEMENTATION OF DATA GUARDS

Figure 3 demonstrates the reference implementation with the Gazebo simulator and PX4. Gazebo is a physical simulator used to perform robotic vehicle simulations. One excellent characteristic of Gazebo is that it has plugins, which are essentially software modules that the user can add or modify. This allows for easy additions and modifications of the simulator. There are several types of plugins, among which are sensor plugins. There is one sensor plugin for each sensor as part of the PX4 integration with Gazebo. For the purposes of this implementation, the GPS plugin was chosen so that perturbations for the GPS signal could be introduced. A simple scheme such as randomly generating a spike in the altitude by generating a random number every ten readings or so is a simple way to perturb the GPS data stream. This emulates a GPS sensor attack or a malfunction in the GPS module. Gazebo sends all sensor data to PX4 through the Mavlink protocol, including the periodic GPS message.

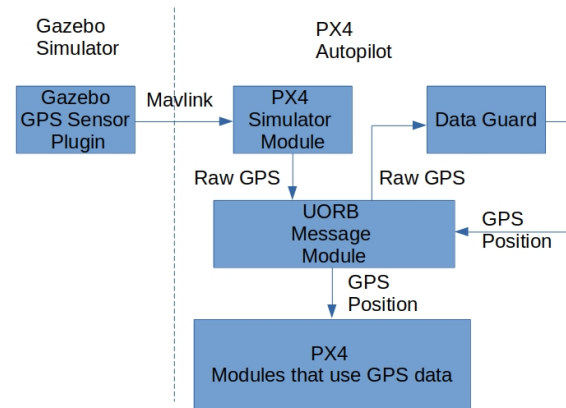


Fig. 3. Reference architecture

The reference implementation is a representative example of the approach based on a popular platform for UAVs and the fact that GPS spoofing is such a common GPS attack. The results show that it is fairly straightforward to introduce a data guard component in a publish-subscriber architecture such as PX4. The same can be done for similar architectures, for example ROS and ROS2 [16], [17]. The performance penalty is minimal if we perform just simple checks. This may not be the case if the data is fairly complex. In this case, our data guard may need to use some ML algorithm or fuzzy logic to achieve its goals. Our experiments showed no degradation of the autopilot's performance, and we expect that in many cases, some spare CPU cycles can be utilized to provide the necessary data protection controllers need.

VI. FUTURE DIRECTIONS

The data guards that we discussed were mainly static as their behavior was specified at design time. However, there may be situations when data fluctuations may require adaptable data guards with more changeable behavior based on ML algorithms. This direction is pretty exciting and also more ambitious. Still, in the era of the ever-increasing use of better hardware and pervasive AI solutions, it is not something that is far from reality. Adaptive behavior has been used in control for a long time, as well as in digital filters. Some of the already established ideas can be applied to complex and variable sensor data with the goal of their online sanitation.

Another possibility is to have an automatic code generator of data guards based on a custom language defining the rules that govern them. This kind of approach can make their implementation even more straightforward and widespread. Automatic code generation can be done from a modeling language or from a data flow language such as Lustre [18]. The objective is to capture the relationships in data processing at a higher level in a fairly representative way and then generate code for the target system. This future direction can be achieved by developing unique tools for the task.

VII. CONCLUSION

This paper demonstrated a flexible architecture for simulating sensor and controller attacks and a mechanism to react to them by introducing data guards. The data guard can be as complex as needed but still be practical for maintaining the real-time response of the system. The approach applies to any sensor and actuator and any controller or module which consumes sensor data. This can include complicated sensors such as image and Lidar sensors. The approach minimizes or eliminates the possibility of affecting the system's stability and normal operation due to sensor data issues. The method is a complementary run-time strategy to data sanitation used during the training of machine learning systems. It makes sensor data more important as part of the set of concerns for robotic systems. Furthermore, the approach applies to the reliability of sensors and malicious modifications of sensor and controller behavior.

REFERENCES

- [1] H. Pearce, S. Pinisetty, P. S. Roop, M. M. Y. Kuo, and A. Ukil, "Smart i/o modules for mitigating cyber-physical attacks on industrial control systems," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 7, pp. 4659–4669, 2020.
- [2] D. de Niz, B. Andersson, and G. Moreno, "Safety enforcement for the verification of autonomous systems," in *Autonomous Systems: Sensors, Vehicles, Security, and the Internet of Everything*, M. C. Dudzik and J. C. Ricklin, Eds., vol. 10643, International Society for Optics and Photonics. SPIE, 2018, pp. 1 – 10. [Online]. Available: <https://doi.org/10.1117/12.2307575>
- [3] H. Choi, W.-C. Lee, Y. Aafer, F. Fei, Z. Tu, X. Zhang, D. Xu, and X. Xinyan, "Detecting attacks against robotic vehicles: A control invariant approach," *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018.
- [4] Y. Zhang and K. Rasmussen, "Detection of electromagnetic interference attacks on sensor systems," in *2020 IEEE Symposium on Security and Privacy (SP)*. Los Alamitos, CA, USA: IEEE Computer Society, may 2020, pp. 1–1. [Online]. Available: <https://doi.ieeecomputersociety.org/10.1109/SP.2020.00001>
- [5] J. Park, R. Ivanov, J. Weimer, M. Pajic, and I. Lee, "Sensor attack detection in the presence of transient faults," in *Proceedings of the ACM/IEEE Sixth International Conference on Cyber-Physical Systems*, ser. ICCPS '15. New York, NY, USA: Association for Computing Machinery, 2015, p. 1–10. [Online]. Available: <https://doi.org/10.1145/2735960.2735984>
- [6] M. T. Leccadito, "A hierarchical architectural framework for securing unmanned aerial systems," 2017.
- [7] A. Allouch, O. Cheikhrouhou, A. Koubaa, M. Khalgui, and T. Abbes, "Mavsec: Securing the mavlink protocol for ardupilot/px4 unmanned aerial systems," *2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC)*, pp. 621–628, 2019.
- [8] J. Zeng, L. T. Yang, M. Lin, H. Ning, and J. Ma, "A survey: Cyber-physical-social systems and their system-level design methodology," *Future Generation Computer Systems*, vol. 105, pp. 1028–1042, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0167739X1630228X>
- [9] M. Wu, H. Zeng, C. Wang, and H. Yu, "Invited: Safety guard: Runtime enforcement for safety-critical cyber-physical systems," in *2017 54th ACM/EDAC/IEEE Design Automation Conference (DAC)*, 2017, pp. 1–6.
- [10] A. Sangiovanni-Vincentelli, W. Damm, and R. Passerone, "Taming dr. frankenstein: Contract-based design for cyber-physical systems," *European journal of control*, vol. 18, no. 3, pp. 217–238, 2012.
- [11] A. Benveniste, B. Caillaud, D. Nickovic, R. Passerone, J.-B. Raclet, P. Reinkemeier, A. Sangiovanni-Vincentelli, W. Damm, T. A. Henzinger, and K. G. Larsen, *Contracts for System Design*, 2018.
- [12] Y. Liu and C. Cunningham, "Software component specification using design by contract," 03 2002.
- [13] M. Sadraey, *Unmanned Aircraft Design: A Review of Fundamentals*, 2017.
- [14] E. Ebeid, M. Skriver, K. H. Terkildsen, K. Jensen, and U. P. Schultz, "A survey of open-source uav flight controllers and flight simulators," *Microprocessors and Microsystems*, vol. 61, pp. 11–20, 2018. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0141933118300930>
- [15] A. Kouba, A. Allouch, M. Alajlan, Y. Javed, A. Belghith, and M. Khalgui, "Micro air vehicle link (mavlink) in a nutshell: A survey," *IEEE Access*, vol. 7, pp. 87 658–87 680, 2019.
- [16] M. Lauer, M. Amy, J.-C. Fabre, M. Roy, W. Excoffon, and M. Stoicescu, "Engineering adaptive fault-tolerance mechanisms for resilient computing on ros," in *2016 IEEE 17th International Symposium on High Assurance Systems Engineering (HASE)*, 2016, pp. 94–101.
- [17] I. Malavolta, G. Lewis, B. Schmerl, P. Lago, and D. Garlan, "How do you architect your robots? state of the practice and guidelines for ros-based systems," in *2020 IEEE/ACM 42nd International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP)*, 2020, pp. 31–40.
- [18] J.-L. Colaço, B. Pagano, and M. Pouzet, "Scade 6: A formal language for embedded critical software development (invited paper)," in *2017 International Symposium on Theoretical Aspects of Software Engineering (TASE)*, 2017, pp. 1–11.

Discovering interactions between applications with log analysis

Lukasz Korzeniowski

Nordea Bank Abp SA, Satamaradankatu 5, FI-00020
 Helsinki, Finland

Email: lukasz.korzeniowski@protonmail.com

Krzysztof Goczyla

Gdańsk University of Technology, Faculty of
 Electronics, Telecommunication and Informatics,

Narutowicza 11/12, Gdańsk, Poland

Email: kris@eti.pg.edu.pl

Abstract—Application logs record the behavior of a system during its runtime and their analysis can provide useful information. In this article, we propose a method of automated log analysis to discover interactions taking place between applications in an enterprise. We believe that such an automated approach can greatly support enterprise architects in building an up-to-date view of a governed system in a modern, fast-paced development environment. Our contribution is the following: we propose a new method for log template generation called SLT (Simple Log Template), we propose a method of extracting knowledge about application interactions from logs, and we validate the proposed methods on a real system running at Nordea Bank. Additionally, we collect statistical information about application logs from the real-life system, based on which we formulate some observations that support our method.

I. INTRODUCTION

ONE of the challenges faced by enterprise architect teams in large organizations is to ensure an up-to-date overview of the systems they are governing. Traditionally, the governance process relies on manual updates to the system representation stored in the architecture repository. However, in the age of microservices and cloud deployments, where significant changes to architecture can take place overnight, this is barely sufficient. There is a clear need for a better, more automated way of maintaining knowledge about systems across an enterprise.

Automation of system knowledge discovery is a big help for enterprise architects, especially in recent times, when new applications are created at an increasing pace and their architecture changes rapidly. Manually updated architecture repositories no longer keep up with the reality of systems deployed to a production environment. This problem is further emphasized by long, heavy, and manual processes of introducing changes to architecture repositories, which do not fit the time-to-market expectations of the business stakeholders. Having a decent representation of current production deployment allows for reasoning about the architecture, detecting non-compliance with internal or external regulations, and helps the development of the enterprise architecture in a planned direction. It also improves building always up-to-date overview of the enterprise system which is beneficial

not only for architects but for developers, analysts, and testers for building a mental model of the system they are working with. One of the potential approaches is to utilize application logs which provide a common, always up-to-date view of applications during their runtime.

Application logs have several compelling advantages as compared to other sources of knowledge about the system (e.g., documentation or source code). Firstly, logging is the most widespread way of tracking system behavior that is present in software development since its beginning. Thanks to that we can assume that nearly every software system, even legacy, implements some form of logging. In the reality of many enterprises, logs may prove to be the only common source of knowledge about systems still running on main-frame platforms. Secondly, logs (which traditionally are stored as text files) are usually human-readable and therefore are suitable for processing by text tools. Additionally, they are a mixture of technical information and narrative. Lastly, logs usually follow the changes in the source code of applications, so they contain the historical aspect of software evolution. All of that makes application logs a rich source of data for various analyses. Because of our focus on automation of knowledge discovery about systems, we are only interested in automated log analysis. According to [1], automated analysis of application logs is a widely studied discipline with growing interest among researchers in recent years. The most obvious usage of log analysis lies in the *operations* area (*intrusion detection, monitoring*) but its potential in reasoning about the *business domain* or the *design* of the software is also explored.

In this paper, we try to derive knowledge about enterprise systems (specifically about interactions between applications) from application logs. Our work fits in the *design/component dependency inference* area of the landscape of automated log analysis proposed by the authors of [1]. We perform experiments on a real-life system from the banking industry, hosted at Nordea Bank. We contribute to the body of knowledge in the following ways:

- we perform statistical analysis of log files of a subset of logs from a real-life system deployed at Nordea Bank,
- we formulate some general observations about the way logs are typically created by developers,

This work was supported by Gdańsk University of Technology

- based on the defined observations, we propose a new method for log template extraction called SLT (Simple Log Template),
- we propose a workflow for automated discovery of application interactions from their logs,
- we verify our method with logs from a real-life system deployed at Nordea Bank.

The remainder of this paper is organized as follows. Section II discusses related work. In Section III, we present a formal definition of the problem. In Section IV, we perform a statistical analysis of application logs from a subset of applications deployed at Nordea Bank and we formulate observations related to how developers place their log statements in enterprise systems. In Section V, we describe our approach for component dependency inference based on log analysis, while in Section VI, we evaluate this method against the real-life system deployed at Nordea Bank. Section VII presents conclusions from our experiments and outlines the future work.

II. RELATED WORK

Component dependency inference using automated log analysis is rather a niche topic, but some notable recent work is worth mentioning. [2] analyzes web service interactions and tries to correlate web service invocations using IP addresses and invocation statuses found in logs. The authors identify two specific types of interactions: composition (one service orchestrates a series of calls to other services) and substitution (service is called as part of an error-handling scenario after a failed call to another service). The authors assume the availability of IP address information in service logs, which (according to [1]) is true mostly for access logs and network logs that may not be available for applications other than web services. Additionally, basing the analysis on IP address correlation may be very challenging in cloud-hosted applications, where services are replicated, and IP addresses can change dynamically. In contrast, our method puts minimum assumptions on log content and bases the analysis on log messages. The authors of [3] perform a statistical analysis of web service logs to identify a correlation between web services. The analysis includes time correlation, call frequency correlation, and analysis of service response times. The authors define three types of web service interactions: dependency on the data source (multiple web services try to access the same shared resource), hierarchical dependency (one service is invoked by another), and serial execution dependency (one service orchestrates a series of invocations). Similarly, as in the previously mentioned work, the focus is on web services and other types of applications are not considered. [4] describes a Bayesian Decision Theory-based approach to the identification of component dependencies. The authors describe each log message with a key and a set of parameters which are determined by some empirical knowledge and common string extraction. The authors also identify some observations related to logging practices that form the foundation for their algorithm: co-oc-

currence observation (logs of dependent services are time-correlated) and correspondence observation (logs of dependent services often contain some identical parameter). The proposed method is validated using the Hadoop dataset. In our work, we take these ideas further by removing any empirical knowledge needed to extract parameters. We also empirically confirm and further extend the authors' observations regarding logging practices based on application logs from a real-life system at Nordea Bank. Furthermore, we validate our method using a dataset of logs from Nordea Bank, which is expected to be less homogeneous than logs of any shared service platform like Hadoop.

Workflow discovery is a similar research area that aims in recovering whole processes from logs. Although it is not in the scope of this paper, workflow discovery is part of our future work and therefore notable work on this related topic is worth mentioning. [8] uses a process mining approach to discover recursive processes from event logs. The authors of [9] propose a method for triaging production failures that analyses service interactions to identify the failed workflows. The authors of [10] present a method for recovery of Communicating Finite State Machine model that represents service interactions. The proposed approach requires, however, users to input knowledge about the structure of a log file for it to be able to be processed.

Our work also aims to identify real-life logging practices applied by software developers. Similar efforts were presented in [5] where different categories of logging statements used by developers are defined. The authors take the source-code perspective analyzing logging practices from the point of view of a single application. [6] presents a statistical analysis of logging practices in open-source projects and [7] repeats this study for java-based open-source projects. The authors analyze factors such as log density, how meaningful log extracts are for bug-fixing, what are the typical changes to logging code, and how often they occur. In our work, we use logs from a real-life system at Nordea Bank to identify higher-level observations regarding logging statements that represent the intent to specifically track interactions between different applications and thus are useful for analyzing application dependencies.

Log template generation is another area of research, related strictly to log analysis, which aims in discovering templates that describe individual lines in log files. Being able to identify static and variable parts of log entries allows for better reasoning about the log content and is considered the basis of any log analysis task. [14] performs a comparative analysis of various log template generation algorithms. Log-Cluster [11] and DRAIN [12] are two of these algorithms that, according to [14], present respectively the lowest accuracy span and the top accuracy levels over the sample data sets. We picked these algorithms as a benchmark for the method proposed by us.

III. PROBLEM STATEMENT

We aim at inferring knowledge about enterprise systems from application logs. The goal is to provide enterprise architects with a good enough representation of the system that reflects the reality of the production environment. The derived model of the enterprise system needs to support architects' activities related to reasoning about the architecture. An example of such activities is data governance which concentrates on aspects like usage of proper data sources by applications, understanding the semantic relationships between data stored and exchanged between applications, or ensuring assumed data flow. Apart from data governance, common enterprise architects' activities concentrate also on ensuring that processes implemented inside the system fulfill both internal and external regulations. These may include measuring process performance or ensuring certain regulatory constraints are obeyed.

We can assume that the set of applications in the enterprise system is known with a high level of confidence. On the other hand, knowledge of application inter-connections (interactions between applications) from the enterprise perspective is where the confidence decreases. Having hundreds of applications deployed in a bank, this confidence is only as good as the diligence of teams in updating a common architecture repository. Furthermore, basing decisions related to the architecture of an enterprise on human declarations, rather than facts, can lead to false conclusions and not-optimal choices. We consider the problem of increasing the confidence of knowledge about the actual application interactions as the core problem in architecture reverse-engineering.

Let $G(S)=(A,C)$ be an undirected graph representing a real system S , where A is a set of applications constituting S and C is a set of edges representing interactions between the applications. We say that applications A_1 and A_2 are interacting with each other if some data exchange takes place between them. Let $L(S)=\{l_1, l_2, \dots, l_n\}$ be a log of activities collected within system S consisting of n messages. We describe each line of the log by a tuple (t, a, m) where t denotes the date and time of log message creation, a represents the application that created the message, m is the actual log message text.

We define the problem of application interaction discovery from logs as finding an approximate graph $G'(S)=(A,C')$, where C' is an approximation of C , based on the system's log $L(S)$. $G'(S)$ is a graph representing an approximation of system S .

The presented problem definition is extendable and allows inference of other architecture properties on top of the applications' interaction graph. For any property of enterprise system $P(S)$, the problem of architecture property discovery from logs can be defined as finding a function F , such that $P'(S)=F(G'(S), L(S))$ approximates $P(S)$. In this paper, we do not cover solutions to the extensions of the core problems, leaving this for further work.

IV. NORDEA BANK DATA SET

Our work presents an experience of applying log analysis to one of the systems at Nordea Bank, which we will further refer to as NDEASYS. For our experiment and proposed method to be as generally applicable as possible, we picked an application with a relatively big integration part, such that logs created by interacting applications are the most representative. We applied the following criteria:

- team diversity – applications built by teams of different sizes, experiences, locations,
- application diversity – we included both dedicated business applications and shared service platforms (e.g., storage or communication services),
- time diversity – applications built in different periods,
- integration diversity – applications communicating using different interfaces and exchange formats.

Nordea Bank does not enforce any strict, centralized rules on how applications should create their logs (for non-regulatory logging), which additionally removes any accidental correlation between logs because of applications being created in the same company.

The NDEASYS1 dataset consists of logs of six applications whose properties are summarized in Table I. The logs were collected over 12 hours of operation on a random business day in an integrated test environment.

We distinguish between three types of applications that output logs in our data set:

- dedicated – application implementing logic specific to a single business domain,
- technical – application with a minimum amount of business logic, usually providing a support function, e.g., data or interface adaptation,
- shared service – application deployed on a shared platform, following a typical workflow for the platform.

We characterized team diversity by the number of developers and number of locations, they were working from. Time diversity was represented by the development period and the duration of application development. The diversity of integration was characterized by the integration styles used by each application (messaging or remote procedure invocation) and message formats used to exchange data with other applications.

We performed an initial analysis of the NDEASYS1 dataset. Fig. 1 presents a histogram of tokens that are present in the logs of each application. The histogram was created using 1000 bins representing the frequency of token occurrence in a log. We made the following observations with regards to all log files:

- in all the logs there is a clear split between a few very common tokens and a lot of very rare tokens (the histogram is right-skewed, see Fig. 1),
- at integration points, application input is commonly logged,
- shared services tend to log only inputs while dedicated and technical applications – both inputs and outputs.

TABLE I.
CHARACTERISTICS OF THE NDEASYS1 DATASET

App	Log size [MB]	Application diversity	Team diversity		Time diversity		Integration diversity	
		Type	Size	No locations	Dev. period	Dev. duration [months]	Integration style	Format
A	730	dedicated	3	2	2020-2022	24	Messaging, RPI	Swift (ISO15022, ISO 20022), JSON
B	40	technical	1	2	2020	1	Messaging	Swift (ISO15022)
C	100	shared service	2	2	2016-2020	48	RPI	JSON
D	0.2	technical	1	2	2020	6	Messaging	Swift (ISO15022)
E	1600	shared service	3	2	2016-2022	72	RPI	JSON
F	3	shared service	3	2	2016-2022	72	RPI	JSON

We interpret these observations from the point of view of the primary reason for logging, which is failure diagnosis. In the case of application integration, a popular practice is to log identifiers of data exchanged between applications. These identifiers are (to large extent) unique values, which explains the big number of very rare tokens appearing in the log as compared to the few frequent tokens that represent the static part of log messages.

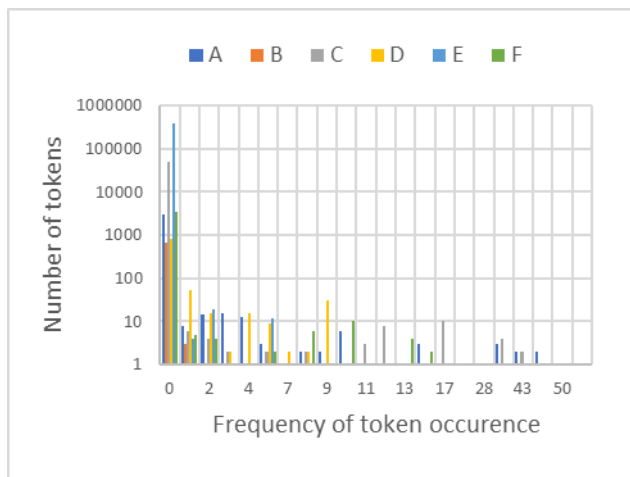


Fig 1. The histogram of token distribution. The horizontal axis represents 1000 bins showing the frequency of token occurrence in each log. The vertical axis is presented on a logarithmic scale and its values describe the number of tokens in each bin.

V. PROPOSED METHOD

Taking the observations presented in Section IV, we propose a method for detecting application interactions from logs leveraging the concept of rare and frequent tokens appearing in log files. Fig. 2. describes the proposed workflow and subsequent sections describe its steps in detail.

A. Log preprocessing

For log files from each application, we perform a minimal level of log preprocessing, which ensures a common view of each log. We introduce minimal assumptions for the content of log files. Each line should contain a *timestamp* and *message* attributes. Additionally, the *source* (the application that created a given line in the log) is determined based on which application the log file belongs to. The process of extraction of the minimum set of information from logs is called log formatting and an example of its output is shown in Fig. 3. In the case of some logs, we also unify data encoding to ensure that logs are comparable between applications. In this step, we perform preprocessing of the *message* attribute by splitting it into individual tokens using a regular expression $[.:A-Za-z0-9_-]+$. Apart from log formatting, we do not apply any application-specific logic requiring expert knowledge. As a result of the log preprocessing step, each log line is described by attributes: *source*, *timestamp*, and *set of tokens*. Example tokens are shown in Table II.

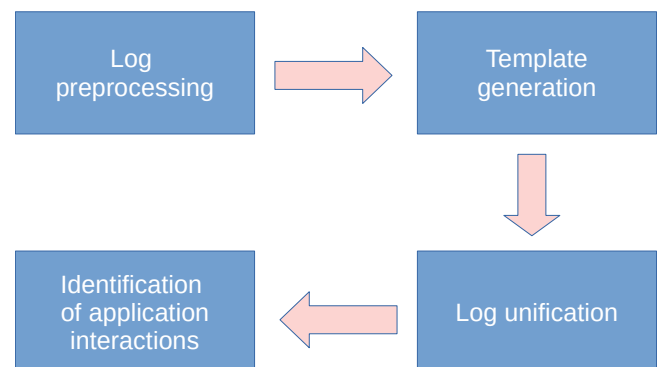


Fig 2. The workflow of log analysis for identification of application interactions.

Raw log

```
timestamp=2022-03-15T08:50:50.030+0100 thread=grpc-
default-executor-441 level=INFO
logger=c.n.t.i.r.q.QueryCallStatsObserver,
operation=QUERY_LATEST_IN_GROUP,
clientId=X, clientLibrary=null, clientVersion=null,
hostName=a01.com, correlationId=4528e974-857c-4623-ab8b-
fe5e45742c41, action=query_start, , domain=Y/Z,
requestCondition={"extracted.id":
"0122318714085000"},
condition={"extracted.id": "0122318714085000"},
groupByFields=[Id], sortFields=[timestamp], limit=0,
payload=true
```

After formatting

```
1647330650034.E,"logger=c.n.t.i.r.q.QueryCallStatsObserver,
operation=QUERY_LATEST_IN_GROUP, clientId=X,
clientLibrary=null, clientVersion=null,
hostName=a01.com, correlationId=969a81d3-8ad8-4a5f-84e8-
0868bfd65ddb, action=query_start, , domain=Y/Z,
requestCondition={"extracted.id": "0122318714085000"},
condition={"extracted.id": "0122318714085000"},
groupByFields=[Id], sortFields=[timestamp], limit=0,
payload=true"
```

Fig 3. The outcome of the log formatting phase. Colors denote the respective fragments in the raw and the formatted log.

B. Template generation

Template generation is the process of determining some sort of pattern for each log line, which distinguishes the static part of the message from the variable parts. Traditionally, this process aims at providing as precise template as possible, which describes the position of each variable element in a log message. We found such an approach not necessary and giving worse results than the relaxed approach proposed in this paper. A more detailed comparison with commonly used template generation methods is provided in Section V.C.

We introduce the SLT (Simple Log Template) method of template generation, which is a relaxed variant of the Log-Cluster approach described in [11]. As compared to the original method, we do not extract the exact pattern but rather focus on splitting the *message* into a *key* (which represents the static part) and *identifiers* (representing the variable part) while entirely disregarding the position of tokens in the *message*. Such an approach serves two purposes. Firstly, it minimizes the number of templates. Secondly, it is better suited for generating templates for log statements with a variable number of repeated tokens. A typical example is logging of service input/output values which are documents exchanged between applications. Such documents, usually expressed in the XML or JSON format, very often are built on top of data schema with repeated occurrences of data items. Moreover, XML or JSON documents cannot be treated as a regular text because, depending on the schema, the order of appearance of their elements can also be variable. In such cases, extraction of the exact pattern would result in a potentially big

number of complex patterns being generated, depending on the data sample.

Our algorithm consists of two phases. In the first one, we cluster log lines using the *set of tokens*. We 1-hot encode [13] each token and then 1-hot encode each log line using the encoding of the tokens it contains and then run the K-means clustering algorithm. An example of token encoding is presented in Table II. The encoding of the sample log line from Fig. 3. is presented in Fig. 4. The optimal number of clusters is determined by using the silhouette method. The idea is, based on the observations in Section IV, that the same 1-hot encoded log lines would differ from one another only on a few positions, which should cause them to be assembled to the same cluster. The clustering process is performed separately for each *source*. This is to ensure that we do not find by accident common templates across different applications. Additionally, it reduces the clustering problem significantly.

TABLE II.
EXAMPLE OF 1-HOT ENCODING OF THE IDENTIFIED TOKENS

Identified token	Word index in dictionary	1-hot encoding
logger	0	[1, 0, ..., 0]
c.n.t.i.r.q.querycallstats observer	1	[0, 1, 0, ..., 0]
operation	2	[0, 0, 1, 0, ..., 0]
query_latest_in_group	38	[0, ..., 0, 1, 0, ..., 0]
clientId	4	[0, 0, 0, 0, 1, 0, ..., 0]
x	39	[0, ..., 0, 1, 0, ..., 0]
clientlibrary	6	[0, 0, 0, 0, 0, 0, 1, 0, ..., 0]

The outcome of the clustering phase is a set of clusters containing specific log lines for each *source*. Each cluster represents a separate logging statement (log template) and the lines that belong to the cluster – instances of that template. During the second phase, we process each cluster individually and extract *identifiers* from the *message* attribute. We count the frequency of each token appearance in the cluster. We then apply a frequency threshold – tokens appearing less frequently than the threshold are considered the *identifiers*. This is a direct utilization of the observation presented in Fig. 1. Both *key* and *identifiers* are represented as a set of tokens – their order is not considered. An example of such a template is presented in Fig. 4. Although this might result in different templates receiving the same key, we rarely found that to be a case in practice. Usually, logging statements are significantly different from one another which ensures the possibility of precisely locating such a logging statement in the source code of the application during failure diagnosis.

In the end, we combine the identifiers from all the clusters into a single set. The outcome of the template generation process is a mapping of log lines to clusters and a set of tokens that are considered identifiers in each application log.

- data loading – an asynchronous process of delivering data to the system; once data is loaded, it is stored for further processing,
- data processing – a scheduled synchronous process occurring every 15 minutes that performs processing of previously delivered data,
- daily reporting – a scheduled synchronous process executed once per day which aggregates the processed data and delivers them to downstream applications.

Although this paper focuses on direct interactions between applications, awareness of the processes helps in configuring our algorithm to find interactions that appear in logs in distant lines.

B. Evaluation method

We use this knowledge about the system to validate our method. As a main measure of accuracy of our method we chose the F1 score and use it to compare the set of edges identified by our algorithm with the reference set of edges presented in Fig. 5. We do not consider the direction of edges.

C. Template generation algorithm performance

A part of our algorithm is a proposal of a new template generation system, focusing specifically on the detection of identifiers in log files. We evaluate our algorithm by comparing its efficiency to popular template generation algorithms: LogCluster [11] and DRAIN [12]. Table III presents the outcome of this comparison for different types of logs: the time of algorithm execution for each data set, the number of identified clusters, precision, recall and F1 score.

Since both LogCluster and DRAIN are more generic algorithms than ours, we need to define objective comparison

criteria. For the sake of algorithm comparison, we decided to consider the number of identified templates, the F1 score of identifier detection and the speed of the algorithm. Both LogCluster and DRAIN produce a set of templates as an output. We unify these templates as regular expressions, where variable parts of each template are transformed into capturing groups. We then process each line of the log file with all the regular expressions and extract the tokens that were captured. LogCluster and DRAIN are not very consistent in what they consider as a token with our algorithm. To remediate that, we post-process each captured group with the same regular expression that is used in our algorithm for token identification. In the end, we apply the same length-based criterion to decide if a token is a valid identifier. The outcome of this post-processing is, for each log file and a set of templates, a list of identifiers found in the file. The list of identifiers is compared to the ground truth derived from the log dataset using our domain knowledge.

Since all the algorithms can be tuned with hyper-parameters, we measured a wide range of parameter values but for the brevity of presentation, we mention only the best score for each of the algorithms.

To extract the ground truth, for each log file, we collected the list of all tokens and the number of their occurrences. We then traversed the list of tokens from the most to the least frequent applying our domain knowledge to remove all tokens which were not valid identifiers. Part of this process was performed automatically. In this phase, we removed tokens based on their length or type (e.g., dates, IP addresses, or cash amounts were not considered valid identifiers, but are easy to filter out using regular expressions). In the second phase, we performed manual filtering by removing to-

TABLE III.
COMPARISON OF TEMPLATE GENERATION METHODS

Algorithm	Log	Time [s]	Clusters	Precision	Recall	F1 Score
DRAIN	A	95.0	31	0.96	1.0	0.98
	B	10.0	335	0.99	0.94	0.96
	C	39.0	6	0.99	1.0	0.99
	D	1.0	82	n/a	n/a	n/a
	E	342	97	1.0	0.86	0.92
	F	1.0	13	0.97	0.99	0.98
LogCluster	A	20.0	6	1.0	0.93	0.96
	B	1.0	28	2.0	0.018	0.03
	C	8.0	2	1.0	0.79	0.88
	D	0.1	3	0.66	1.0	0.79
	E	34.0	5	1.0	0.85	0.91
	F	0.1	3	1.0	0.95	0.97
SLT	A	185.0	17	0.95	1.0	0.97
	B	1.0	2	0.71	0.99	0.83
	C	46.0	19	0.94	0.99	0.97
	D	0.1	3	0.6	1.0	0.75
	E	380	19	1.0	1.0	1.0
	F	2.0	2	0.76	1.0	0.86

kens that were valid domain-related words. This process was performed for the most frequent tokens (with the number of occurrences higher than 1).

LogCluster is the fastest algorithm among the three. It outperforms the rest by an order of magnitude. It also notes the lowest F1 score, especially for log B. It tends to keep the precision very high with lower recall values. Further analysis of false positives returned by this algorithm shows that it often includes frequent tokens, which are business terms appearing in the log. LogCluster returns fewer and more generic clusters than others.

DRAIN shows the best overall results in terms of F1 score with by far the highest number of clusters returned. The high number of clusters is a result of two aspects: DRAIN not being designed for processing multi-line log entries (similar as LogCluster) and not coping well with log entries of variable length. While the first deficiency is easy to overcome, the second poses a challenge for general use for enterprise-wide log analysis. A typical case for log entries with variable content is logging system inputs, which often come in the form of XML/JSON documents with repeated elements. In such cases, DRAIN extracts each log entry with a given length to a separate template. For long log entries, templates are often very big and hard to process. This finding is in line with the conclusions from the real-life DRAIN application presented in [14].

The SLT approach proposed by us is not far from DRAIN in terms of F1 score, with similar timing performance but outputting the number of clusters that is much more on par with the reality. It copes well with log entries of variable length. One deficiency, that results in lowered precision rates, is falsely identifying rare numbers (e.g., cash amounts) as identifiers. With the approach we have taken, they cannot be distinguished from the true identifiers. This problem needs to be handled in the subsequent processing steps that utilize our algorithm. Overall, we think SLT is a reasonable all-around approach for identifying identifiers in log files of diverse format and content.

D. Results

We ran a series of experiments with our approach using different time windows. For each run of the algorithm, we collected the discovered application interactions, together with the set of identifiers that are found in the logs of both interacting applications within the time window. An example of such output is presented in Table IV, and the respective approximate graph $G'(S)$ is shown in Fig. 6. In Table IV, falsely identified interactions are marked in red.

TABLE IV.
EXAMPLE OF DISCOVERED APPLICATION INTERACTIONS

Interaction	Number of occurrences	Example identifier
(E, A)	47828	5435ab2142314bsw
(C, A)	2156	43543612124
(F, A)	1314	2353518
(F, C)	874	200.0000

(D, A)	77	abhgswe0053
(D, E)	14	2022-01-21
(B, A)	7	basdewe2xyz

In some cases, it is hard to distinguish between an accidental correlation of token values and actual but rare interactions. For example, the (F, C) interaction has a higher number of occurrences than (D, A) but the discovered identifiers actually denote the number of records per second processed by services F and C. Such an accidental correlation of tokens representing data of low variety (e.g. small numbers or dates) is the main source of score deterioration in our method.

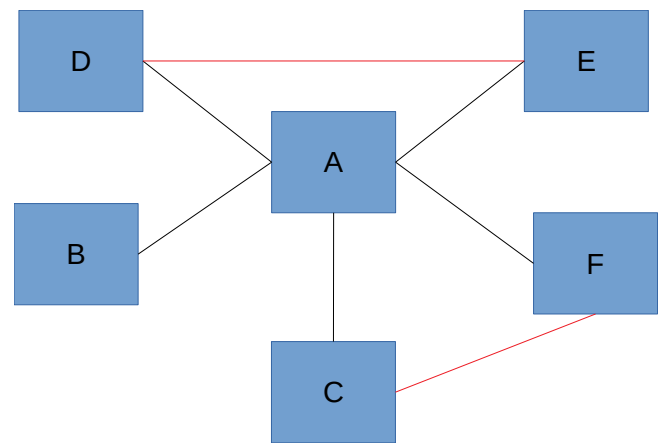


Fig 6. An example of graph G' output that represents the approximation of the system, at a window size equal to 20 minutes. Edges marked with red represent falsely discovered application interactions.

We found that applying a proper identifier length criterion allows for maximizing the F1 score of the result. Table V shows the maximum scores achieved for each size of the time window. The maximum overall F1 score of our algorithm was 83%, reached for a time window of 20 minutes which was long enough to detect the interaction between the asynchronous data loading process and the scheduled data processing process. Longer time windows allow us to identify distant correlations at the cost of the decrease in precision (the longer the time window, the higher chance of accidental correlation occurring). At the current stage, our approach tends to discover short-time-distance interactions with high precision. The confidence of interaction detection decreases when searching for distant correlations. This is one of the aspects that are subject to improvement in our future work.

TABLE V.
RESULTS OF OUR METHOD FOR DIFFERENT WINDOW SIZES

Window size [ms]	Number of discovered interactions	Precision	Recall	F1 Score
200	3	1	0.6	0.75
1000	3	1	0.6	0.75

5000	4	0.75	0.6	0.67
10000	4	0.75	0.6	0.67
60000	5	0.8	0.8	0.8
1200000	7	0.71	1	0.83

VII. CONCLUSIONS

In this paper, we presented an approach to discovering the interactions between applications in enterprise systems. We validated our approach with a real-life system deployed at Nordea Bank. Our method could achieve the 83% F1 score of the identified interactions and is a good base for further extension. The biggest challenge is distinguishing rare, actual interactions from accidental data correlation, which we will address in our further study.

As part of our approach, we proposed the SLT method for discovering templates for log entries. We compared this method with other common approaches and found that it provides good-enough F1 score while being able to handle variable log entries, which is one of the main deficiencies of the other methods. We find SLT a good candidate for a template generation method in a general use case when we do not know the exact profile of the logs we are analyzing.

Working on a real-life system allowed us also to provide statistics and conclusions about logs from various types of applications. We found some common patterns of logging activities performed by developers when working with integrated systems which stem from the practical need of developers to be able to perform failure diagnosis. The main conclusion in this area is that logging data identifiers is a common practice in the industry. These observations allow introduction of simplifying assumptions for the general problem of discovering system properties from application logs and help better focus our future work on the problem.

Our future work will focus on two areas: 1) improvement of the current method and 2) working on its extensions to extract other properties of enterprise systems. We see increasing the precision of the identifier detection as the main improvement that would positively influence the F1 score of the overall method. This requires the development of a better way to check if an identifier is valid to decrease the level of accidental correlation of non-identifier tokens. Another area of focus is the performance of our method, improvement of which would allow for the analysis of larger log samples covering longer periods. That would open the possibility to identify very distant interactions (e.g., related to monthly-reporting processes), or validate the method using a larger system.

As for the method extensions, our goal is to discover the fragments of UML diagrams describing the working system. That requires providing means of discovering the system's properties such as:

- processes the system is running (control flow) – scenarios of interactions between applications with their frequency and timing,

- data flow – how data is passed across applications and what are their origins,
- semantic relationships between data – the mapping of data processed by different applications to find a common data dictionary (or even the data model).

We believe that such an overview of the running system would provide enterprise architects with the necessary tools to centrally validate various properties of the system and plan respective actions accordingly.

ACKNOWLEDGMENT

This paper was written in cooperation with the Nordea Bank, which provided the log dataset and an overview of the systems that were subject to this study.

REFERENCES

- [1] L. Korzeniowski and K. Goczyla, "Landscape of Automated Log Analysis: A Systematic Literature Review and Mapping Study," *IEEE Access*, vol. 10, pp. 21892–21913, 2022.
- [2] H. Labbaci, B. Medjahed, and Y. Aklouf, "Learning interactions from web service logs," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10439 LNCS, no. August, pp. 275–289, 2017.
- [3] E. U. Aktas, M. C. Calpur, U. U. Yildirim, and E. Yildirim, "Inferring dependencies among web services with predictive and statistical analysis of system logs," *CEUR Workshop Proc.*, vol. 2291, no. December, pp. 235–244, 2018.
- [4] J. G. Lou, Q. Fu, Y. Wang, and J. Li, "Mining dependency in distributed systems through unstructured logs analysis," *Oper. Syst. Rev.*, vol. 44, no. 1, pp. 91–96, 2010.
- [5] Q. Fu *et al.*, "Where do developers log? an empirical study on logging practices in industry," 2014, pp. 24–33.
- [6] D. Yuan, S. Park and Y. Zhou, "Characterizing logging practices in open-source software," 2012 34th International Conference on Software Engineering (ICSE), 2012, pp. 102–112, doi: 10.1109/ICSE.2012.6227202.
- [7] B. Chen and Z. M. (Jack) Jiang, "Characterizing logging practices in Java-based open source software projects – a replication study in Apache Software Foundation," *Empir. Softw. Eng.*, vol. 22, no. 1, pp. 330–374, Feb. 2017.
- [8] M. Leemans, W. M. P. Van Der Aalst, and M. G. J. Van Den Brand, "Recursion aware modeling and discovery for hierarchical software event log analysis," *25th IEEE Int. Conf. Softw. Anal. Evol. Reengineering, SANER 2018 - Proc.*, vol. 2018-March, no. March, pp. 185–196, 2018.
- [9] G. Qi, W. T. Tsai, W. Li, Z. Zhu, and Y. Luo, "A cloud-based triage log analysis and recovery framework," *Simul. Model. Pract. Theory*, vol. 77, no. August 2020, pp. 292–316, 2017.
- [10] I. Beschastnikh, Y. Brun, M. D. Ernst, and A. Krishnamurthy, "Inferring models of concurrent systems from logs of their behavior with CSight," 2014, pp. 468–479.
- [11] R. Vaarandi and M. Piheigas, "LogCluster - A data clustering and pattern mining algorithm for event logs," *Proc. 11th Int. Conf. Netw. Serv. Manag. CNSM 2015*, pp. 1–7, 2015.
- [12] P. He, J. Zhu, Z. Zheng, and M. R. Lyu, "Drain: An Online Log Parsing Approach with Fixed Depth Tree," *Proc. - 2017 IEEE 24th Int. Conf. Web Serv. ICWS 2017*, pp. 33–40, 2017.
- [13] J. T. Hancock and T. M. Khoshgoftaar, "Survey on categorical data for neural networks," *J. Big Data*, vol. 7, no. 1, 2020.
- [14] J. Zhu *et al.*, "Tools and Benchmarks for Automated Log Parsing," *Proc. - 2019 IEEE/ACM 41st Int. Conf. Softw. Eng. Softw. Eng. Pract. ICSE-SEIP 2019*, pp. 121–130, 2019.

Joint 42nd IEEE Software Engineering Workshop and 9th International Workshop on Cyber-Physical Systems

THE IEEE Software Engineering Workshop (SEW) is the oldest Software Engineering event in the world, dating back to 1969. The workshop was originally run as the NASA Software Engineering Workshop and focused on software engineering issues relevant to NASA and the space industry. After the 25th edition, it became the NASA/IEEE Software Engineering Workshop and expanded its remit to address many more areas of software engineering with emphasis on practical issues, industrial experience and case studies in addition to traditional technical papers. Since its 31st edition, it has been sponsored by IEEE and has continued to broaden its areas of interest.

One such extremely hot new area are Cyber-physical Systems (CPS), which encompass the investigation of approaches related to the development and use of modern software systems interfacing with real world and controlling their surroundings. CPS are physical and engineering systems closely integrated with their typically networked environment. Modern airplanes, automobiles, or medical devices are practically networks of computers. Sensors, robots, and intelligent devices are abundant. Human life depends on them. CPS systems transform how people interact with the physical world just like the Internet transformed how people interact with one another.

The joint workshop aims to bring together all those researchers with an interest in software engineering, both with CPS and broader focus. Traditionally, these workshops attract industrial and government practitioners and academics pursuing the advancement of software engineering principles, techniques and practices. This joint edition will also provide a forum for reporting on past experiences, for describing new and emerging results and approaches, and for exchanging ideas on best practice and future directions.

TOPICS

The workshop aims to bring together all those with an interest in software engineering. Traditionally, the workshop attracts industrial and government practitioners and academics pursuing the advancement of software engineering principles, techniques and practice. The workshop provides a forum for reporting on past experiences, for describing new and emerging results and approaches, and for exchanging ideas on best practice and future directions.

Topics of interest include, but are not limited to:

- Experiments and experience reports

- Software quality assurance and metrics
- Formal methods and formal approaches to software development
- Software engineering processes and process improvement
- Agile and lean methods
- Requirements engineering
- Software architectures
- Design methodologies
- Validation and verification
- Software maintenance, reuse, and legacy systems
- Agent-based software systems
- Self-managing systems
- New approaches to software engineering (e.g., search based software engineering)
- Software engineering issues in cyber-physical systems
- Real-time software engineering
- Safety assurance & certification
- Software security
- Embedded control systems and networks
- Software aspects of the Internet of Things
- Software engineering education, laboratories and pedagogy
- Software engineering for social media

TECHNICAL SESSION CHAIRS

- **Bowen, Jonathan**, Museophile Ltd., United Kingdom
- **Hinchey, Mike** (Lead Chair), Lero-the Irish Software Engineering Research Centre, Ireland
- **Szmuc, Tomasz**, AGH University of Science and Technology, Poland
- **Zalewski, Janusz**, Florida Gulf Coast University, United States

PROGRAM COMMITTEE

- **Ait Ameer, Yamine**, Toulouse Institute for Research in Computer Science, France
- **Banach, Richard**, University of Manchester, United Kingdom
- **Challenger, Moharram**, University of Antwerp, Belgium
- **Cicirelli, Franco**, DIMES Università della Calabria, Italy
- **Gomes, Luis**, Universidade NOVA de Lisboa, Portugal
- **Gracanin, Denis**, Virginia Tech, USA

- **Havelund, Klaus**, Jet Propulsion Laboratory, USA
- **Hsiao, Michael**, Virginia Tech, USA
- **Karaduman, Burak**, University of Antwerp, Belgium
- **Katwijk, Jan van**, TU Delft, Netherlands
- **Pullum, Laura**, Oak Ridge National Laboratory, USA
- **Sekerinski, Emil**, McMaster University, Canada
- **Sojka, Michal**, Czech Technical University in Prague, Czech Republic
- **Trybus, Leszek**, Rzeszow University of Technology, Poland
- **Vardanega, Tullio**, University of Padua, Italy
- **Velev, Miroslav**, Aries Design Automation, USA

Software Sentiment Analysis using Deep-learning Approach with Word-Embedding Techniques

Venkata Krishna Chandra Mula¹

Dept. CSIS
Andhra University College of Engineering
krishnacmula@gmail.com

Lalita Bhanu Murthy³

Department of Computer Science & Information Systems
BITS Pilani Hyderabad Campus
bhanu@hyderabad.bits-pilani.ac.in

Lov Kumar²

Department of Computer Science & Information Systems
BITS Pilani Hyderabad Campus
lovkumar@hyderabad.bits-pilani.ac.in

Prof. Aneesh Krishna⁴

Curtin University
A.Krishna@curtin.edu.au

Abstract—Sentiment Analysis in the Software Engineering community aims to make the development and maintenance of software a better experience by helping provide code and library suggestions, defect-related comments for source code, etc. The manual finding of sentiment-based comments may be an inaccurate prediction and a time-consuming process. Automating the sentiment analysis process by leveraging Machine Learning models can benefit software professionals by giving them insights into other developers and feelings about software products, libraries, development, and maintenance tasks at a glance. This study aims to develop software sentiment prediction models based on comments by (1) identifying the best embedding techniques to represent the word of the comments, not just as a number but as a vector in n-dimensional space (2) finding the best sets of vectors using different features selection techniques (3) finding the best methods to handle the class imbalance nature of the data, and (4) finding the best architecture of deep-learning for the training of models. The developed models are validated using 5-fold cross-validation with four different performance parameters: accuracy, AUC, recall, and precision on three different datasets. The experimental finding shows that the models developed using the word embeddings with feature selection using Deep Learning classifiers on balanced data can significantly predict the underlying sentiments of textual comments.

Keywords—Sentiment Analysis, Software Engineering Tasks, Word Embedding, Feature Selection, Data Imbalance, SMOTE, Deep Learning classifiers

I. INTRODUCTION

SENTIMENT Analysis can be used to gather the opinions and feelings of the consumers regarding social and political opinions, brand loyalty, etc. Sentiment Analysis utilizes natural language processing and machine learning algorithms to draw out textual data's mood, opinions, and feelings. The texts can be product reviews, posts on social media, messages on chat boards, answers to questions on a question-answering website, commit messages by developers, etc [1]. Software developers can utilize the benefits of sentiment analysis to assist them in their development and maintenance activities. They can be well informed about whether a particular software, technology, or tutorial is appropriate for their purposes. Sentiment Analysis can distinguish feedback as being either positive or negative, which helps in development decisions. The application of

sentiment analysis is to find Software professionals' sentiments and mindsets, and it can be approached in two different ways: The first and most straightforward way is to sit down face to face with the software developers, glean insights into their mindset, and evaluate their mood and feelings. Unfortunately, this is a very time-intensive and tedious process to incorporate. So, the other approach is preferred. The other approach uses sentiment analysis to discern the mood and feelings of software developers, from the product reviews, feedback forms, commit messages, etc. An effort is made to identify the positive and negative sentiments of developers. So, we have worked to create a predictive model leveraging natural language processing techniques and machine learning algorithms, to predict and detect the exhibited sentiments, moods, and feelings effectively and efficiently. The datasets are obtained from user's App Reviews, and issues tracked and managed using JIRA, and user's comments and messages on the Stack Overflow platform [1]. For the machine to better understand and analyze the text to improve the predictive model's performance, we must represent words as vectors [2][3].

Word embedding techniques do this by representing words as vectors in an n-dimensional space. This provides a numeric representation to the words, which allows them to be used as input to Machine Learning Models, and it also preserves its syntactic and semantic integrity so that words that are used similarly have similar vector representations. In this study, we use six Word Embedding Techniques to vectorize the textual data, which are Term Frequency and Inverse Document Frequency (TF-IDF), Skip-Gram (SKG), Continuous Bag of Words (CBOW), Global Vectors for Word Representations (GLOVE), Fast Text (FST) and Google News Word to Vector (GW2V) [3]. After applying the word embeddings, we obtain a multitude of features for the data, many of which will be ineffective in the predictive model. To obtain the subset of important features that are to be used as input to the model, we apply six Feature Selection Techniques, namely Principal Components Analysis, Gain Ratio Attribute Eval, Classifier Attribute Eval, Info Gain Attribute Eval, OneR Attribute Eval, and Analysis of Variance (ANOVA) [4]. Analyzing the data after applying the Feature Selection Techniques, it is clear that the data suffer from the class imbalance problem, which

occurs when the number of samples in each class is not the same. If not corrected, this can negatively affect the predictive performance of the model. So, the Synthetic Minority Oversampling Technique (SMOTE) and the Borderline Synthetic Minority Oversampling Technique (Borderline-SMOTE) are applied to balance the data. After we balance the data, we need to compare and evaluate the performance of the different techniques we have applied. To achieve this, we use eight different deep learning classifiers, which are applied by varying the number of Hidden Layers and Dropout Layers. The application of Deep Learning Classifiers to the models developed using different Word Embedding Techniques can help determine the models that can accurately and effectively predict the underlying sentiment in textual data, which can be convenient for a broad scope of Software Development and Maintenance activities. This study also aims to find the Word Embeddings, Feature Selection, and Data Sampling Techniques that provide the most optimal results.

The remainder of the paper is laid out as follows: Section 2 presents a literature review on software sentiment analysis and various word embedding approaches. Section 3 describes the experimental dataset collection as well as the various machine learning algorithms used. The research methodology is described in Section 4 using an architecture framework. In Section 5, the results of the experiments, along with their analysis, are presented. Section 6 shows a comparison of models created using various word-embedding approaches, sets of features, and machine learning models. Finally, Section 7 summarizes the information provided and offers directions for further research.

II. RELATED WORK

There are many methods to acquire features from textual data. Term Frequency and Inverse Document Frequency (TF-IDF) have been used by Rajni Jindal et al. to obtain features from defect descriptions. They've used a Radial Basis function of the Neural Network to classify the defect reports. Based on tangible evidence, they've established that the model predicted high severity defects with significant accuracy and efficiency [5]. Sari and Siahaan have also leveraged Term Frequency and Inverse Document Frequency (TF-IDF) to extract features from defect descriptions. They've applied the InfoGain Feature Selection technique to obtain the set of relevant features. They've built severities prediction models with the assistance of Support Vector Machine to predict severity levels of defects [6]. Sentiment Analysis of Software Engineering Tasks has tremendous potential, but pre-trained models don't accurately predict sentiments in Software Engineering Tasks. Bin Lin et al. applied Deep Learning techniques to an enormous dataset consisting of 40k manually labeled sentences, which were sourced from Stack Overflow. Despite determining all the text's sentiments, it resulted in low accuracy levels and, ultimately, poor results. In a comparison between Stanford CoreNLP and Stanford CoreNLP SO, it was determined that the Stanford CoreNLP SO was a better influence on Sentiment Analysis tools than the Stanford CoreNLP. Another conclusion reached by Bin Lin et al. was that Sentiment Analysis tools that are not specifically trained for Software Engineering data yield disappointing results on Software Engineering datasets [1].

Biswas et al. used software domain-specific Word Embed-

ding learned from Stack Overflow in an attempt to improve the performance of the predictive model. The impact on the performance of Sentiment Analysis tools using Domain-specific Word Embedding and Generic Word Embedding, trained using Google news, were compared. The conclusion reached was that the Generic Word Embedding was better than the Domain-specific Word embedding. Biswas et al. also found that oversampling or a combination of oversampling and undersampling achieves a jump in performance in the handling of compact Software Engineering datasets[2]. R Malhotra et al. have attempted to develop Software Bug Classification (SBC) models that can identify "low", "moderate," and "high" impact levels on Software Bugs. The levels were indicated based on Maintenance Effort (ME), Change Impact (CI), or a product of both. The data is sourced from the changelogs in Google's GIT Repository. Data preprocessing is performed, and the SBC models are developed using six classification techniques. The study assessed three predictors, which were obtained from text mining. After evaluation, it was found that the performance of the combined SBC model showed higher accuracy than the ME or CI SBC models. They also found that the accuracy of the "high" category was superior to that of the other categories [7].

R Malhotra et al. have worked to find out if resampling methods applied to software defect data improve performance. They have used datasets sourced from the Defect Collection and Reporting tool (DCRS) and performed data preprocessing and applied three different resampling methods, and evaluated their performance. The performance of the developed models is evaluated using seven performance measures, accuracy, precision, sensitivity, specificity, G-Mean, and AUC. They have concluded that the application of resampling methods to the maintainability prediction models can accurately predict the minority class [5]. R Malhotra et al. have also attempted to find the effects on the performance of Software Defect Prediction Models after applying resampling techniques. They have applied six oversampling and four undersampling methods to rectify the class imbalance problem. Examining the evaluators, which are: Sensitivity, GMean, Balance, and AUC values, it was found that there was an evident improvement in the values of the evaluators when data resampling methods were applied to the Software Defect Prediction models [8].

Dr. Lov Kumar et al. have worked to automate the process of determining the severity level of a defect in the software. Defect descriptions in the form of text have been tokenized using seven different word embedding techniques. The obtained features are further pruned to achieve an optimal set of relevant features using three different Feature Selection techniques. These features, plagued by the Class Imbalance Problem, have been rid of it by using the Synthetic Minority Oversampling Technique (SMOTE). The performance of the Word Embedding is evaluated using eleven different classifiers. Dr. Lov Kumar et al. have successfully used Word Embeddings, Feature Selection, and Synthetic Minority Oversampling Technique (SMOTE) to assemble a predictive model capable of assigning a severity level to defect descriptions [9]. SentiStrength-SE, which was proposed by Islam et al., achieved 73.85% precision and 85% recall. They call attention to issues commonly faced by Sentiment Analysis Tools, some of which are: Domain-specific meaning of words, Context-sensitive variations in meanings of words, Difficulties in dealing with negation, Sentimental words in copy-pasted content,

Difficulty in dealing with irony and sarcasm, and Wrong detection of proper nouns [3].

III. STUDY DESIGN

This section presents the details regarding various design settings used for this research.

A. Experimental Dataset

The study uses three different experiential datasets to validate our proposed framework. These datasets are used by many software researchers for sentiment analysis [1][2]. The primary objective is to explore different types of embedding, feature selection, data sampling, and different variants of deep-learning on these datasets to predict the sentiments of software engineers. Figure 2 shows the number of positive and negative sentiments for the considered datasets. From Figure 2, we observed that the number of positive sentiments for stack overflow is much higher than negative sentiments. The unequal distribution of data leads to a class imbalance problem.

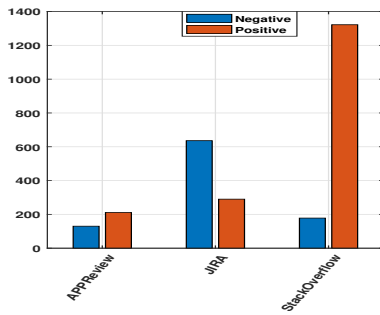


Fig. 2: Data-Sets

B. Training of Models from Imbalanced Data Set:

After analysis of the data, it becomes quite evident that the data is suffering from a class imbalance problem, i.e., the number of samples in each class is not the same. So, the balancing of data is required to improve the predictive ability of the developed Sentiment Analysis Models. We have performed Synthetic Minority Oversampling Technique (SMOTE) and Borderline Synthetic Minority Oversampling Technique (Borderline-SMOTE) on each dataset to balance the data [10].

C. Word Embedding:

The textual data of the dataset is to be expressed as vectors in relation to each other. Six different word embedding techniques including Term Frequency and Inverse Document Frequency (TF-IDF), Continuous Bag of Words (CBOW), Skip-Gram (SKG), Global Vectors for Word Representation (GLOVE), Google news Word to Vector (GW2V), fasttext (FST) have been applied on the dataset. These techniques were used to represent the textual data as a vector in an n-dimensional space. We have also removed any and all stopwords, bad symbols, and spaces before applying the word embedding techniques. These will now be used to develop models to determine the sentiment of Software Engineering Tasks [9][2].

D. Feature Selection Techniques

The features vectors extracted from word-embedding are used as an input, so, the performance also depends upon the optimization of important features. To extract the important features from the existing set of vectors, we have used six different Feature Selection Techniques such as: Analysis of Variance (ANOVA) is used to find feature having capability to differentiate positive and negative sentiment, correlation attribute evaluation (CORR_ATR) is used to remove highly correlated features, Principal Components Analysis (PCA) is used to find new value of uncorrelated features, Gain ratio, information gain, and OneR are used to rank the features and select best features for sentiment analysis [4][11].

E. Classification Technique:

In this study, we have used eight deep learning models, which use K-Fold Cross-Validation with a k value of 5. We have separated the data into training and testing data subsets. An input layer with a number of neurons equal in quantity to the number of features of the input data is present in every single deep learning model. The models are all constituted of Dense and Dropout layers. The Dense layer's neurons receive inputs from all the neurons present in the previous layer. The Dropout layer's neurons are randomly selected. The dropout value used in this study is 0.2. The output layer has a single neuron that corresponds to the binary classification of either functional or non-functional requirements, and it uses a sigmoid activation function, unlike the other layers, which use the

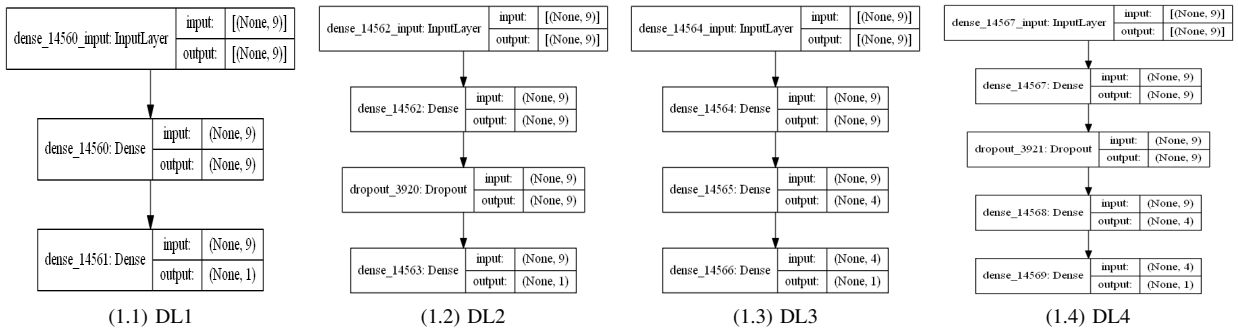


Fig. 1: Deep Learning Architecture

Rectified Linear Activation function (ReLU) as the activation function. Adam is the optimizer used to train the models, with the loss function being the Binary Cross entropy. The number of hidden layers is increased for the other four models. Figure 1 demonstrates the architectures of the models (DL1, DL2, DL3, and DL4). The parameters used to validate the models are: batch size = 30, Dropout = 0.2, and epochs = 100.

IV. RESEARCH METHODOLOGY

The pictorial representation of the proposed framework is provided in Figure 3. We first extract the textual documents from three different SE repositories, i.e., issue trackers, i.e., JIRA issue comments, Stack Overflow discussions contain questions and answers and user reviews on mobile apps using app stores so that their corresponding data may be analyzed. After finding these textual documents, six different types of word-embedding techniques have been used to represent text documents as numerical vectors. Each embedding uses a different way to represent words for text documents with a real-valued vector. The values of these vectors are closer in the vector space for similar words. Next, we have used two different types of sampling techniques, such as: SMOTE and BLSMOTE to handle the class imbalanced nature of datasets. In the next step, we have applied different feature selection techniques to select the best combination of relevant features. The ANOVA test is used to remove insignificant features, PCA is used to remove high correlation between features and find new values of features, gain ratio, information gain, and OneR is used to rank features and select the top best features, and finally, correlation analysis is used to remove highly correlated features. After finding the right sets of features, we have used different variants of deep-learning techniques to train software sentiment, and analysis models. The trained models are validated with a 5-fold cross-validation method, and the performance of these models is compared with the help of four different performance parameters: accuracy, precision, recall, and AUC.

V. EMPIRICAL RESULTS AND ANALYSIS

The primary objective of this work is to analyze the performance of the developed software sentiment analysis models using different variants of deep-learning, word-embedding techniques, features selection techniques, and data sampling techniques with the purpose of investigating how different contexts can impact their effectiveness. The proposed models are validated with three software-related datasets, namely mobile app reviews, Stack Overflow discussions, and JIRA issue comments. Finally, the predictive ability of these models is evaluated using different performance parameters such as accuracy, AUC, precision, and recall. AUC is considered the primary parameter for the model's performance because of its capability to provide good findings in case of imbalanced nature of data. Tables I and II show the performance of models in terms of Precision, Recall, accuracy, and AUC for the AppReview dataset using different variants of deep-learning and feature selection techniques with original data and sampled data. The results for other combinations are similar. The high value of AUC (≥ 0.7) in Tables I and II suggested that the proposed models have the capability to predict the current state of sentiment analysis for software engineering. In the majority of cases, the precision, recall, and AUC values are higher than 0.8. Also, the information present in Tables I and II suggest that the models trained on sampled data have a better ability to predict sentiment as compared to original data. Another finding from Tables I and II is that the models trained on selected sets of features have a high value of precision, recall, and AUC as compared to all features.

VI. COMPARATIVE ANALYSIS

In this section, we have compared the performance of different word embedding techniques, class balancing approaches, feature selection strategies, and deep-learning techniques, which are used for developing sentiment prediction models using Box-plot diagrams and descriptive statistics. We have also performed the Friedman test to find statistical significance differences between different techniques. The hypothesis used to achieve our objective is mentioned below:

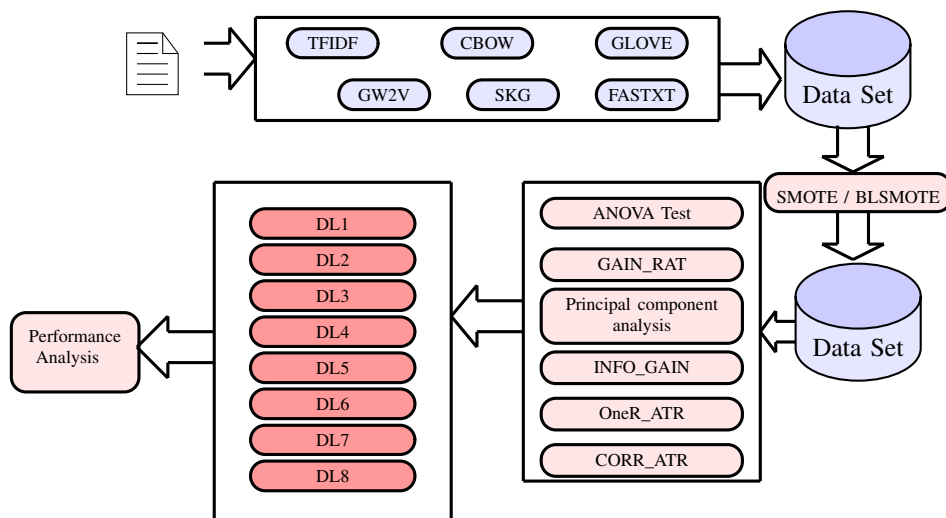


Fig. 3: Framework of proposed work

TABLE I: AppReviews :Precision and Recall

	Precision								Recall							
	DL1	DL2	DL3	DL4	DL5	DL6	DL7	DL8	DL1	DL2	DL3	DL4	DL5	DL6	DL7	DL8
ORGDATA(OD)																
TFIDF	0.85	0.84	0.84	0.84	0.85	0.83	0.83	0.84	0.90	0.87	0.85	0.87	0.87	0.87	0.88	0.88
CBOW	0.64	0.72	0.76	0.68	0.67	0.76	0.64	0.64	0.95	0.86	0.63	0.80	0.90	0.81	0.92	0.98
SKG	0.81	0.82	0.83	0.8	0.83	0.85	0.82	0.81	0.7	0.86	0.81	0.86	0.87	0.75	0.77	0.83
GLOVE	0.86	0.86	0.85	0.85	0.88	0.84	0.85	0.87	0.88	0.85	0.86	0.88	0.87	0.84	0.88	0.84
GW2V	0.84	0.83	0.85	0.85	0.87	0.85	0.86	0.85	0.87	0.88	0.87	0.89	0.79	0.86	0.88	0.85
FASTXT	0.74	0.77	0.77	0.76	0.73	0.73	0.73	0.76	0.73	0.68	0.73	0.69	0.82	0.72	0.66	0.69
ORGDATA(ANOVA)																
TFIDF	0.87	0.87	0.86	0.87	0.87	0.84	0.87	0.84	0.89	0.91	0.91	0.91	0.88	0.92	0.89	0.91
CBOW	0.62	0.62	0.71	0.68	0.66	0.66	0.62	0.64	1.00	1.00	0.88	0.89	0.96	0.90	1.00	0.98
SKG	0.81	0.83	0.8	0.86	0.85	0.85	0.82	0.83	0.82	0.83	0.82	0.68	0.83	0.80	0.83	0.83
GLOVE	0.87	0.85	0.84	0.87	0.84	0.88	0.88	0.88	0.84	0.84	0.86	0.83	0.85	0.83	0.8	0.83
GW2V	0.89	0.86	0.87	0.86	0.89	0.87	0.84	0.87	0.86	0.89	0.88	0.85	0.88	0.87	0.87	0.88
FASTXT	0.74	0.76	0.76	0.76	0.76	0.75	0.76	0.75	0.84	0.86	0.77	0.82	0.80	0.80	0.75	0.72
ORGDATA(OneR_ATR)																
TFIDF	0.82	0.82	0.83	0.83	0.78	0.73	0.72	0.62	0.93	0.93	0.93	0.94	0.97	0.97	0.97	1.00
CBOW	0.62	0.62	0.64	0.66	0.64	0.70	0.62	0.62	1.00	1.00	0.99	0.96	0.98	0.91	1.00	1.00
SKG	0.71	0.71	0.71	0.70	0.73	0.73	0.62	0.68	0.84	0.91	0.87	0.87	0.82	0.84	1.00	0.90
GLOVE	0.84	0.82	0.83	0.82	0.83	0.74	0.80	0.64	0.85	0.85	0.82	0.82	0.85	0.89	0.87	0.95
GW2V	0.85	0.86	0.85	0.83	0.83	0.7	0.82	0.79	0.83	0.84	0.84	0.86	0.87	0.91	0.84	0.87
FASTXT	0.63	0.69	0.71	0.69	0.68	0.67	0.63	0.64	0.88	0.82	0.81	0.78	0.77	0.83	0.99	0.81
SMOTE(OD)																
TFIDF	0.87	0.92	0.90	0.92	0.92	0.90	0.92	0.91	0.94	0.94	0.93	0.94	0.94	0.94	0.93	0.94
CBOW	0.66	0.68	0.78	0.71	0.68	0.81	0.67	0.73	0.8	0.98	0.92	0.97	0.98	0.90	1.00	0.96
SKG	0.81	0.86	0.89	0.84	0.84	0.91	0.86	0.84	0.66	0.94	0.90	0.91	0.91	0.85	0.91	0.91
GLOVE	0.88	0.92	0.93	0.93	0.93	0.93	0.92	0.93	0.87	0.94	0.90	0.91	0.94	0.93	0.90	0.92
GW2V	0.94	0.95	0.93	0.92	0.93	0.94	0.93	0.94	0.96	0.94	0.95	0.92	0.95	0.95	0.95	0.95
FASTXT	0.79	0.77	0.81	0.75	0.77	0.78	0.73	0.77	0.87	0.87	0.85	0.90	0.89	0.90	0.92	0.85
SMOTE(ANOVA)																
TFIDF	0.88	0.92	0.89	0.91	0.90	0.89	0.92	0.89	0.97	0.96	0.97	0.96	0.97	0.97	0.97	0.96
CBOW	0.67	0.67	0.68	0.67	0.67	0.67	0.67	0.68	1.00	1.00	0.94	1.00	1.00	0.99	1.00	0.95
SKG	0.78	0.85	0.90	0.85	0.85	0.88	0.82	0.87	0.95	0.90	0.88	0.92	0.91	0.90	0.92	0.90
GLOVE	0.87	0.93	0.93	0.93	0.92	0.93	0.92	0.94	0.94	0.94	0.93	0.92	0.92	0.92	0.95	0.94
GW2V	0.92	0.92	0.93	0.93	0.92	0.94	0.93	0.94	0.96	0.95	0.96	0.95	0.95	0.95	0.93	0.96
FASTXT	0.79	0.78	0.81	0.78	0.80	0.83	0.83	0.84	0.84	0.92	0.87	0.92	0.91	0.88	0.89	0.87
SMOTE(OneR_ATR)																
TFIDF	0.83	0.84	0.85	0.83	0.83	0.73	0.77	0.72	0.94	0.96	0.95	0.94	0.94	0.97	0.96	0.97
CBOW	0.67	0.67	0.68	0.69	0.73	0.67	0.67	0.68	1.00	0.99	0.95	0.97	0.95	1.00	1.00	0.96
SKG	0.70	0.70	0.71	0.70	0.68	0.69	0.67	0.68	0.97	0.98	0.95	0.97	0.98	0.98	1.00	0.93
GLOVE	0.80	0.79	0.83	0.83	0.79	0.71	0.7	0.72	0.89	0.90	0.86	0.89	0.90	0.95	0.99	0.97
GW2V	0.78	0.79	0.79	0.80	0.80	0.83	0.79	0.74	0.88	0.88	0.86	0.90	0.90	0.88	0.88	0.90
FASTXT	0.68	0.71	0.74	0.72	0.69	0.75	0.67	0.73	0.98	0.97	0.87	0.95	0.98	0.87	0.99	0.90
BLSMOTE(OD)																
TFIDF	0.89	0.92	0.92	0.94	0.93	0.92	0.93	0.92	0.92	0.92	0.91	0.93	0.93	0.9	0.94	0.92
CBOW	0.66	0.67	0.80	0.73	0.67	0.76	0.67	0.76	0.8	0.99	0.92	0.96	1.00	0.93	1.00	0.95
SKG	0.79	0.80	0.87	0.83	0.87	0.88	0.82	0.86	0.75	0.94	0.92	0.91	0.84	0.90	0.84	0.90
GLOVE	0.9	0.93	0.92	0.93	0.93	0.93	0.91	0.92	0.93	0.93	0.92	0.94	0.92	0.93	0.94	0.93
GW2V	0.95	0.95	0.95	0.89	0.96	0.95	0.87	0.95	0.95	0.94	0.95	0.95	0.94	0.94	0.95	0.94
FASTXT	0.77	0.74	0.76	0.72	0.73	0.75	0.69	0.73	0.87	0.90	0.85	0.84	0.83	0.84	0.91	0.87
BLSMOTE(ANOVA)																
TFIDF	0.87	0.92	0.89	0.91	0.91	0.89	0.91	0.90	0.94	0.95	0.95	0.95	0.95	0.95	0.95	0.96
CBOW	0.67	0.67	0.67	0.67	0.67	0.67	0.67	0.69	1.00	1.00	0.95	1.00	1.00	0.98	1.00	0.98
SKG	0.74	0.83	0.89	0.83	0.79	0.85	0.83	0.90	0.97	0.90	0.90	0.89	0.92	0.91	0.88	0.87
GLOVE	0.87	0.93	0.95	0.95	0.93	0.95	0.92	0.92	0.94	0.94	0.92	0.93	0.92	0.91	0.94	0.93
GW2V	0.94	0.95	0.95	0.94	0.94	0.96	0.93	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.96	0.95
FASTXT	0.74	0.74	0.83	0.78	0.77	0.86	0.73	0.81	0.90	0.92	0.83	0.85	0.91	0.78	0.94	0.86
BLSMOTE(OneR_ATR)																
TFIDF	0.80	0.81	0.79	0.81	0.81	0.75	0.76	0.73	0.94	0.95	0.95	0.95	0.95	0.97	0.98	0.98
CBOW	0.67	0.67	0.73	0.70	0.67	0.68	0.67	0.67	1.00	0.99	0.94	0.97	1.00	0.99	1.00	0.99
SKG	0.70	0.69	0.69	0.68	0.69	0.69	0.67	0.68	0.95	0.97	0.95	0.97	0.95	0.97	1.00	0.98
GLOVE	0.77	0.75	0.80	0.76	0.74	0.70	0.69	0.75	0.92	0.96	0.9	0.95	0.97	0.95	0.98	0.91
GW2V	0.74	0.76	0.75	0.78	0.76	0.78	0.73	0.77	0.89	0.88	0.89	0.90	0.90	0.90	0.94	0.87
FASTXT	0.67	0.67	0.70	0.70	0.67	0.69	0.67	0.68	0.98	0.98	0.95	0.94	0.99	0.96	0.99	0.98

- Word-Embedding Techniques: Null Hypothesis:** There is no significant difference between the models trained by using features extracted by different embedding techniques.
- Feature Selection Techniques: Null Hypothesis:** There is no significant difference between the models trained by using selected sets of features using different feature selection techniques and all features.
- Data Sampling Techniques: Null Hypothesis:** There is no significant difference between the models trained on sampled data and original data.
- Deep-Learning Techniques: Null Hypothesis:** There is no significant difference between the models trained using different variants of deep-learning techniques.

TABLE II: AppReviews: Accuracy and AUC

	Accuracy								AUC							
	DL1	DL2	DL3	DL4	DL5	DL6	DL7	DL8	DL1	DL2	DL3	DL4	DL5	DL6	DL7	DL8
ORGDATA(OD)																
TFIDF	83.87	81.52	80.65	82.11	82.11	80.94	81.82	82.4	0.89	0.85	0.85	0.85	0.84	0.83	0.84	0.82
CBOW	64.22	70.67	64.81	64.52	65.69	72.43	62.46	64.52	0.63	0.7	0.73	0.69	0.71	0.78	0.64	0.67
SKG	71.55	79.77	77.42	77.71	80.65	76.54	75.07	77.13	0.81	0.87	0.85	0.85	0.88	0.86	0.83	0.86
GLOVE	83.58	81.82	82.11	82.7	84.16	80.65	83.28	82.4	0.91	0.9	0.89	0.9	0.91	0.89	0.9	0.89
GW2V	81.82	81.52	82.7	83.58	79.77	82.4	83.28	81.52	0.91	0.91	0.89	0.91	0.9	0.87	0.91	0.87
FASTXT	67.45	67.45	69.21	67.16	69.5	66.28	63.64	66.86	0.76	0.77	0.75	0.75	0.74	0.74	0.71	0.77
ORGDATA(ANOVA)																
TFIDF	84.46	86.22	85.63	85.92	84.75	84.75	85.04	84.16	0.91	0.9	0.87	0.89	0.88	0.85	0.88	0.85
CBOW	61.88	62.46	70.38	67.74	67.45	64.52	62.17	64.52	0.59	0.75	0.73	0.74	0.74	0.65	0.69	0.52
SKG	77.13	78.59	75.95	73.61	80.35	78.59	78.01	78.59	0.84	0.86	0.84	0.85	0.85	0.86	0.84	0.85
GLOVE	82.4	81.23	81.52	81.82	80.35	82.7	80.65	82.4	0.89	0.88	0.88	0.88	0.88	0.89	0.88	0.88
GW2V	84.46	84.16	84.16	81.82	85.92	84.16	81.52	84.46	0.92	0.92	0.9	0.91	0.92	0.9	0.9	0.89
FASTXT	71.85	75.07	70.97	72.73	71.85	71.55	70.09	67.74	0.73	0.82	0.78	0.79	0.79	0.77	0.78	0.75
ORGDATA(OneR_ATR)																
TFIDF	83.58	83.58	84.16	84.75	80.94	76.25	75.07	61.88	0.85	0.84	0.83	0.84	0.81	0.77	0.76	0.47
CBOW	61.88	61.58	64.81	67.45	64.22	70.38	61.88	61.88	0.68	0.79	0.71	0.78	0.77	0.78	0.75	0.48
SKG	68.92	71.26	69.5	68.62	70.38	70.97	61.88	67.45	0.74	0.75	0.72	0.71	0.73	0.76	0.6	0.67
GLOVE	80.65	79.47	78.59	78.01	79.77	73.61	78.3	64.52	0.88	0.88	0.86	0.88	0.85	0.77	0.85	0.55
GW2V	80.65	81.52	80.65	80.65	80.65	70.67	78.3	77.71	0.88	0.88	0.86	0.87	0.87	0.75	0.85	0.85
FASTXT	61	65.98	68.04	64.52	63.64	63.93	63.64	59.53	0.64	0.71	0.71	0.69	0.69	0.64	0.69	0.54
SMOTE(OD)																
TFIDF	86.2	90.14	88.17	90.14	90.86	89.25	90.14	89.43	0.92	0.92	0.89	0.92	0.91	0.9	0.92	0.9
CBOW	58.96	68.28	77.6	71.68	67.74	79.21	66.67	73.66	0.62	0.74	0.79	0.82	0.74	0.81	0.66	0.81
SKG	66.67	85.3	85.66	82.62	82.62	84.41	83.69	82.26	0.77	0.9	0.91	0.87	0.88	0.91	0.89	0.86
GLOVE	83.51	90.68	89.25	89.43	91.4	90.86	87.81	90.14	0.9	0.94	0.95	0.94	0.96	0.93	0.94	0.93
GW2V	92.83	92.47	91.94	89.96	91.58	92.47	92.11	92.29	0.97	0.97	0.95	0.95	0.94	0.94	0.94	0.94
FASTXT	75.81	74.01	76.52	73.12	74.55	76.34	72.4	73.3	0.81	0.77	0.82	0.76	0.77	0.79	0.73	0.78
SMOTE(ANOVA)																
TFIDF	88.53	91.22	90.14	91.04	91.04	90.5	92.11	88.89	0.92	0.94	0.91	0.94	0.94	0.91	0.94	0.9
CBOW	66.67	66.67	67.03	66.67	66.67	66.31	66.67	67.38	0.57	0.65	0.68	0.68	0.7	0.57	0.67	0.59
SKG	78.49	82.8	85.13	83.69	82.97	85.66	81.36	84.77	0.84	0.88	0.92	0.88	0.88	0.9	0.86	0.9
GLOVE	86.38	91.4	90.86	90.14	89.07	90.68	91.22	92.11	0.92	0.95	0.95	0.94	0.94	0.95	0.96	0.95
GW2V	91.94	91.58	92.65	92.29	91.76	92.83	90.86	93.01	0.97	0.97	0.96	0.97	0.96	0.97	0.96	0.94
FASTXT	73.84	77.42	77.96	77.06	78.32	79.39	80.29	80.11	0.73	0.79	0.83	0.81	0.81	0.81	0.82	0.83
SMOTE(OneR_ATR)																
TFIDF	82.8	84.95	84.95	83.69	83.15	74.01	78.67	72.76	0.83	0.82	0.84	0.83	0.82	0.7	0.79	0.69
CBOW	66.67	66.67	67.03	68.28	72.94	66.67	66.67	66.85	0.49	0.63	0.75	0.78	0.78	0.49	0.4	0.5
SKG	70.07	70.43	70.79	69.89	67.74	69.89	66.67	65.77	0.69	0.71	0.7	0.71	0.71	0.61	0.69	0.55
GLOVE	77.42	77.6	78.49	80.29	77.42	70.43	70.43	73.48	0.83	0.83	0.84	0.85	0.83	0.62	0.59	0.69
GW2V	75.63	76.16	75.81	78.49	78.67	79.57	76.52	72.4	0.81	0.83	0.81	0.83	0.83	0.84	0.82	0.74
FASTXT	68.1	70.97	71.15	72.58	69.71	72.22	67.2	70.97	0.64	0.7	0.71	0.72	0.67	0.72	0.69	0.69
BLSMOTE(OD)																
TFIDF	87.28	89.78	88.17	91.22	90.86	88.53	91.22	89.25	0.92	0.92	0.91	0.93	0.93	0.91	0.92	0.91
CBOW	58.6	67.56	79.21	73.12	67.03	75.45	66.67	76.52	0.6	0.72	0.82	0.77	0.71	0.77	0.54	0.8
SKG	70.07	80.65	85.3	81.9	81.54	84.95	76.7	83.69	0.78	0.9	0.91	0.88	0.88	0.9	0.86	0.89
GLOVE	87.99	90.68	89.25	91.22	90.14	90.5	89.78	89.96	0.93	0.94	0.93	0.94	0.95	0.94	0.92	0.93
GW2V	93.55	92.83	93.55	89.25	93.01	92.65	87.28	92.83	0.97	0.97	0.97	0.94	0.97	0.96	0.92	0.95
FASTXT	73.84	71.51	72.4	67.74	68.64	70.79	67.03	70.25	0.77	0.74	0.75	0.73	0.71	0.74	0.69	0.74
BLSMOTE(ANOVA)																
TFIDF	86.92	91.22	89.25	90.68	90.86	89.25	91.04	90.14	0.91	0.93	0.9	0.93	0.92	0.89	0.92	0.91
CBOW	66.67	66.49	66.31	66.85	66.67	67.03	66.67	69	0.58	0.67	0.67	0.68	0.64	0.56	0.65	0.66
SKG	75.63	81.36	86.02	80.65	78.49	83.87	79.93	84.59	0.84	0.88	0.91	0.88	0.87	0.89	0.85	0.91
GLOVE	86.92	90.86	91.04	92.47	90.5	90.68	90.86	89.78	0.93	0.95	0.94	0.95	0.94	0.93	0.94	0.94
GW2V	92.47	93.19	93.73	92.47	92.65	93.91	92.83	93.55	0.97	0.97	0.97	0.96	0.96	0.97	0.97	0.95
FASTXT	72.22	73.3	77.6	74.55	76.16	76.88	73.12	77.06	0.73	0.8	0.83	0.81	0.79	0.82	0.8	0.81
BLSMOTE(OneR_ATR)																
TFIDF	80.65	82.26	79.75	82.08	81.72	76.7	78.32	74.37	0.75	0.75	0.75	0.76	0.76	0.72	0.76	0.63
CBOW	66.67	66.31	73.12	70.25	66.67	68.82	66.67	67.38	0.53	0.64	0.71	0.75	0.72	0.53	0.58	0.54
SKG	69.18	69	68.64	67.74	68.82	68.82	66.67	67.56	0.64	0.64	0.65	0.62	0.71	0.63	0.56	0.55
GLOVE	76.7	75.63	78.67	76.88	75.27	69	69.35	73.84	0.79	0.79	0.8	0.82	0.8	0.6	0.74	0.74
GW2V	71.86	74.01	72.94	75.99	74.73	76.34	73.12	74.19	0.78	0.79	0.78	0.8	0.8	0.8	0.77	0.77
FASTXT	67.2	66.67	70.07	68.64	66.49	68.28	66.49	68.28	0.58	0.66	0.67	0.66	0.67	0.57	0.58	0.51

A. Word-Embedding

In this work, six different types of word embedding approaches such as TFIDF, CBOW, GLOVE, GW2V, SKG, and FASTXT have been used to find the numerical vectors of software text comments. To find the best embedding approach, we exploited performance evaluators- Accuracy, Precision, AUC, and Recall, which are computed for models trained

by taking the above embedding techniques as input and trained using different variants of deep-learning with 5-fold cross-validation techniques on sampled as well as original data. Figure 4 visually depicts the model's ability to predict sentiments using different word-embedding techniques, and Table III depicts descriptive statistics of different embedding in terms of accuracy, AUC, precision, and Recall. From Figure 4, it is visible that the models trained by taking numerical

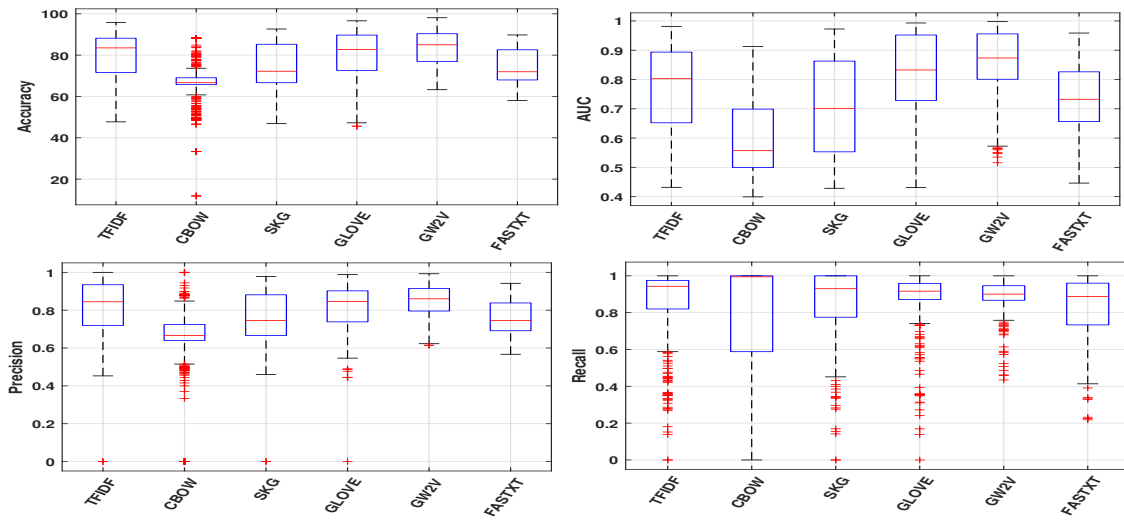


Fig. 4: Performance Box-Plot Diagram: Performance of Different Word Embedding

vectors computed using google word to vector (GW2V) have better capability to predict sentiment as compared to other embeddings. The models trained using GW2V achieved 0.86 average AUC, 0.89 average Recall, 0.85 average precision, and 84.03 average accuracy. Similarly, From Table III, we observed that the models trained using CBOW have the worst performance with 0.60 average AUC, 0.78 average Recall, 0.66 average precision, and 67.49 average accuracy.

TABLE III: Descriptive Statistics: Different Word Embedding

	Min	Max	Mean	Q2	Q1	Q3
Accuracy						
TFIDF	47.72	95.83	80.39	83.51	71.58	88.20
CBOW	11.87	88.13	67.49	66.67	65.73	69.05
SKG	46.86	92.66	75.30	72.22	66.67	85.26
GLOVE	45.60	96.62	81.46	82.70	72.63	89.71
GW2V	63.29	98.11	84.03	85.02	76.92	90.38
FASTXT	58.06	89.78	74.66	71.96	68.00	82.59
Precision						
TFIDF	0.00	1.00	0.82	0.84	0.72	0.93
CBOW	0.00	1.00	0.66	0.67	0.64	0.72
SKG	0.00	0.98	0.77	0.75	0.67	0.88
GLOVE	0.00	0.99	0.82	0.85	0.74	0.90
GW2V	0.61	0.99	0.85	0.86	0.80	0.91
FASTXT	0.57	0.94	0.76	0.74	0.69	0.84
Recall						
TFIDF	0.00	1.00	0.87	0.94	0.82	0.98
CBOW	0.00	1.00	0.78	0.99	0.59	1.00
SKG	0.00	1.00	0.87	0.93	0.78	1.00
GLOVE	0.00	1.00	0.89	0.92	0.87	0.96
GW2V	0.43	1.00	0.89	0.90	0.87	0.95
FASTXT	0.22	1.00	0.84	0.89	0.73	0.96
AUC						
TFIDF	0.43	0.98	0.78	0.80	0.65	0.89
CBOW	0.40	0.91	0.60	0.56	0.50	0.70
SKG	0.43	0.97	0.70	0.70	0.55	0.86
GLOVE	0.43	0.99	0.82	0.83	0.73	0.95
GW2V	0.52	1.00	0.86	0.87	0.80	0.96
FASTXT	0.45	0.96	0.74	0.73	0.66	0.83

Table IV shows the mean ranks using the Freidman test for the various word embedding techniques. We have evaluated

the considered null hypothesis at 0.05 with five degrees of freedom on four different performance parameters such as accuracy, recall, precision, and AUC. The lower value of mean rank represents the best word-embedding techniques for sentiment analysis of software engineering comments. According to information present in Table IV, the models trained using different embedding techniques are significantly different. Similarly, according to information present in Table IV, the models trained using GW2V have a lower mean rank, i.e., 1.91 for accuracy, 1.92 precision, 3.61 recall, and 1.37 AUC representing that the developed models have better prediction capability as compared to other embedding techniques.

B. Feature Selection

In this work, six different types of features selection techniques: significant features calculation using ANOVA test, un-correlated sets of features using PCA, best sets of features using the gain ratio(GAIN_RAT), information gain(INFO_GAIN), oneR attribute evaluation (OneR_ATR), correlation attribute selection (CORR_ATR) have been used to find the best combination of relevant features for software engineering sentiment analysis. We exploited performance evaluators- Accuracy, Precision, AUC, and Recall to find the best feature selection technique that gives us the best sets of features for models trained using different variants of deep-learning with 5-fold cross-validation techniques on sampled as well as original data. Figure 5 visually depicts the model’s ability to predict sentiments using different feature selection techniques and Table V depicts descriptive statistics of different feature selection techniques in terms of accuracy, AUC, precision, and Recall. From Figure 4, it is quite evident that the models trained by taking significant sets of features using the ANOVA test have better capability to predict sentiment as compared to other feature selection techniques. The models trained using ANOVA features achieved 0.85 average AUC, 0.88 average recall, 0.84 average precision, and 83.21 average accuracy. Similarly, From Table V, we observed that the models trained using features selection from OneR_ATR have

TABLE IV: Friedman test : Mean Rank

	Accuracy	AUC	Precision	Recall
DL				
DL1	4.69	4.48	4.60	4.80
DL2	3.83	3.20	4.03	4.27
DL3	3.49	3.15	3.39	5.17
DL4	3.46	3.07	3.68	4.69
DL5	4.12	3.70	4.32	4.04
DL6	5.01	5.62	4.79	4.78
DL7	5.62	6.03	5.67	3.65
DL8	5.78	6.75	5.51	4.60
	$P < 0.05$	$P < 0.05$	$P < 0.05$	$P < 0.05$
Word-Embedding				
TFIDF	2.74	3.25	2.43	3.52
CBOW	5.27	5.58	5.39	3.07
SKG	4.17	4.49	4.16	3.15
GLOVE	2.60	2.23	2.74	3.46
GW2V	1.91	1.37	1.92	3.61
FASTXT	4.32	4.07	4.37	4.19
	$P < 0.05$	$P < 0.05$	$P < 0.05$	$P < 0.05$
Feature Sets				
OD	2.44	2.09	2.39	4.02
ANOVA	2.27	1.72	2.21	3.77
PCA	4.49	4.72	4.74	3.59
GAIN_RAT	4.83	4.93	4.80	4.10
INFO_GAIN	4.56	4.46	4.48	4.15
OneR_ATR	4.91	5.35	4.90	4.10
CORR_ATR	4.50	4.73	4.48	4.28
	$P < 0.05$	$P < 0.05$	$P < 0.05$	$P < 0.05$
OD and SMOTE				
ORGDATA	1.69	2.28	1.83	2.34
SMOTE	2.02	1.73	1.88	1.87
BLSMOTE	2.29	1.98	2.29	1.79
	$P < 0.05$	$P < 0.05$	$P < 0.05$	$P < 0.05$

the worst performance with 0.70 average AUC, 0.84 average Recall, 0.75 average precision, and 74.39 average accuracy.

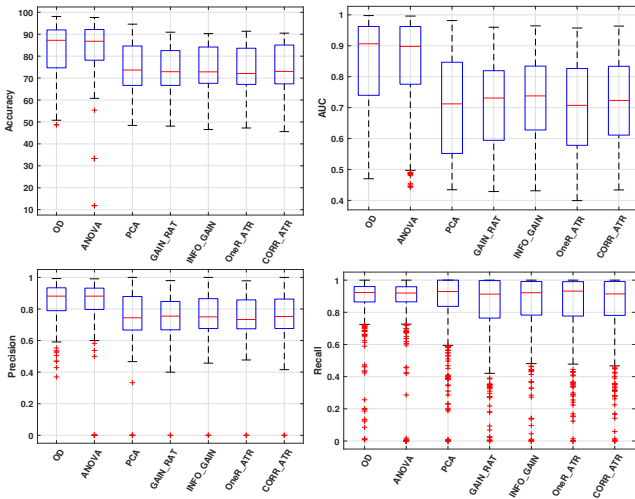


Fig. 5: Performance Box-Plot Diagram: Performance of Different Sets of Features

Table IV shows the mean ranks using the Friedman test for the various feature selection techniques. We have evaluated the considered null hypothesis at 0.05 with six degrees of freedom on four different performance parameters such as accuracy, recall, precision, and AUC. According to information present in Table IV, the models trained using different feature selection techniques are significantly different. Similarly, information

TABLE V: Descriptive Statistics: Different Sets of Features

	Min	Max	Mean	Q2	Q1	Q3
Accuracy						
OD	48.66	98.11	83.40	87.24	74.69	91.96
ANOVA	11.87	97.64	83.21	86.80	78.16	92.17
PCA	48.43	94.60	74.83	73.67	66.67	84.63
GAIN_RAT	48.11	90.93	74.51	72.91	66.67	82.54
INFO_GAIN	46.54	90.28	75.08	72.85	67.62	84.16
OneR_ATR	47.25	91.36	74.39	72.10	67.14	83.61
CORR_ATR	45.60	90.50	75.12	73.05	67.37	85.06
Precision						
OD	0.37	0.99	0.85	0.88	0.79	0.93
ANOVA	0.00	0.99	0.84	0.88	0.80	0.93
PCA	0.00	1.00	0.75	0.74	0.67	0.88
GAIN_RAT	0.00	0.98	0.76	0.75	0.67	0.85
INFO_GAIN	0.00	1.00	0.76	0.75	0.68	0.87
OneR_ATR	0.00	0.98	0.75	0.73	0.67	0.86
CORR_ATR	0.00	1.00	0.76	0.75	0.68	0.86
Recall						
OD	0.01	1.00	0.88	0.92	0.86	0.96
ANOVA	0.00	1.00	0.88	0.92	0.87	0.96
PCA	0.00	1.00	0.86	0.93	0.84	1.00
GAIN_RAT	0.00	1.00	0.84	0.91	0.76	1.00
INFO_GAIN	0.00	1.00	0.85	0.92	0.78	0.99
OneR_ATR	0.00	1.00	0.84	0.93	0.78	0.99
CORR_ATR	0.00	1.00	0.84	0.91	0.78	0.99
AUC						
OD	0.47	1.00	0.83	0.91	0.74	0.96
ANOVA	0.44	1.00	0.85	0.90	0.78	0.96
PCA	0.43	0.98	0.71	0.71	0.55	0.85
GAIN_RAT	0.43	0.96	0.71	0.73	0.59	0.82
INFO_GAIN	0.43	0.96	0.73	0.74	0.63	0.83
OneR_ATR	0.40	0.96	0.70	0.71	0.58	0.83
CORR_ATR	0.43	0.96	0.72	0.72	0.61	0.83

present in Table IV shows that the models trained by taking selected significant features using ANOVA as an input have a lower mean rank, i.e., 2.27 for accuracy, 2.21 precision, 3.77 recall, and 1.72 AUC, represent that the developed models have better prediction capability as compared to other features selection techniques.

C. Classification Techniques

The sentiment prediction models for software engineering comments are trained using different variants of deep-learning techniques with a 5-fold cross-validation approach. These trained models' capability is compared using performance parameters such as accuracy, precision, recall, and AUC. Figure 6 depicts the box-plot diagrams of different performance parameters for the models trained using different variants of deep learning. Table VI shows the descriptive statistics in terms of Mean, Median, Min, Max, Q1, and Q3 for different deep-learning techniques. It can be seen from Figure 6 and Table VI that the models trained using DL2, DL3, DL4, and DL5 have similar average values of AUC, i.e., 0.78. Similarly, the DL8 classifier produces models with a minimum average AUC of 0.67.

In this paper, we have also compared the effectiveness of different variants of deep learning using the Friedman test with a significance level of 0.05 and six degrees of freedom on four different performance parameters such as accuracy, recall, precision, and AUC. Table IV shows the mean ranks using the Friedman test for different variants of deep learning. According to the results of the Friedman test, we observed

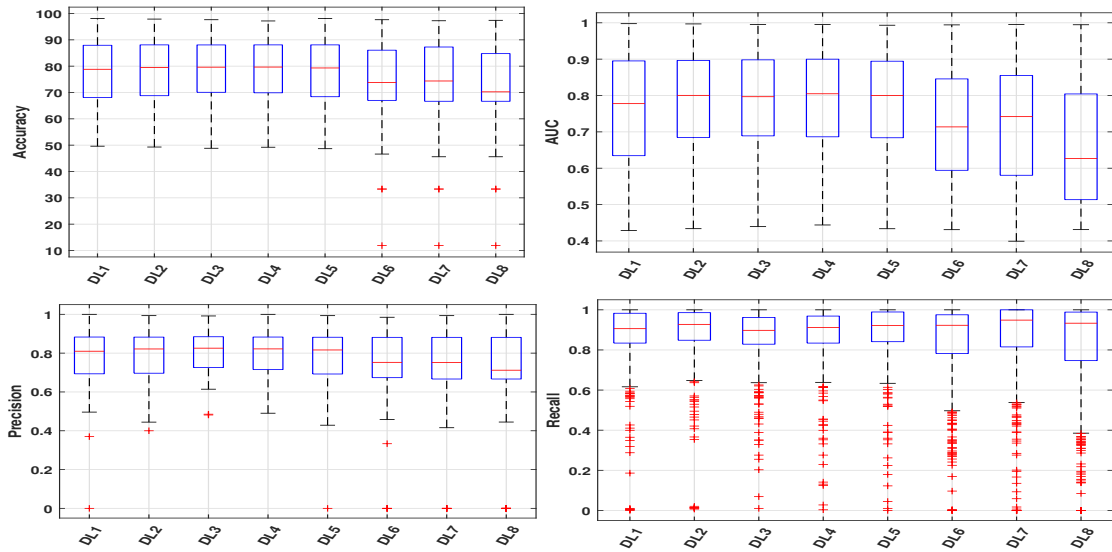


Fig. 6: Performance Box-Plot Diagram: Different Variants of Deep Learning

TABLE VI: Descriptive Statistics: Different Variants of DL

	Min	Max	Mean	Q2	Q1	Q3
Accuracy						
DL1	49.61	98.11	77.97	78.81	68.09	87.93
DL2	49.29	97.88	78.96	79.53	68.79	88.09
DL3	48.82	97.64	79.19	79.62	70.07	88.07
DL4	49.21	97.17	79.07	79.66	69.89	88.09
DL5	48.66	98.11	78.69	79.32	68.40	88.09
DL6	11.87	97.64	75.30	73.82	67.00	86.07
DL7	11.87	97.25	75.41	74.37	66.67	87.28
DL8	11.87	97.41	73.15	70.25	66.67	84.77
Precision						
DL1	0.00	1.00	0.79	0.81	0.69	0.88
DL2	0.40	0.99	0.80	0.82	0.70	0.88
DL3	0.48	0.99	0.81	0.83	0.73	0.88
DL4	0.49	1.00	0.81	0.82	0.72	0.88
DL5	0.00	0.99	0.80	0.82	0.69	0.88
DL6	0.00	0.98	0.76	0.75	0.67	0.88
DL7	0.00	0.99	0.76	0.75	0.67	0.88
DL8	0.00	1.00	0.73	0.71	0.67	0.88
Recall						
DL1	0.00	1.00	0.86	0.91	0.83	0.98
DL2	0.01	1.00	0.88	0.93	0.85	0.99
DL3	0.01	1.00	0.86	0.90	0.83	0.96
DL4	0.00	1.00	0.87	0.91	0.83	0.97
DL5	0.00	1.00	0.88	0.92	0.84	0.99
DL6	0.00	1.00	0.83	0.92	0.78	0.98
DL7	0.00	1.00	0.85	0.95	0.82	1.00
DL8	0.00	1.00	0.81	0.93	0.75	0.99
AUC						
DL1	0.43	1.00	0.76	0.78	0.63	0.90
DL2	0.43	1.00	0.78	0.80	0.68	0.90
DL3	0.44	1.00	0.78	0.80	0.69	0.90
DL4	0.44	1.00	0.78	0.80	0.69	0.90
DL5	0.43	0.99	0.78	0.80	0.68	0.89
DL6	0.43	0.99	0.72	0.71	0.59	0.85
DL7	0.40	1.00	0.73	0.74	0.58	0.86
DL8	0.43	0.99	0.67	0.63	0.51	0.80

that the performance of software sentiment prediction models significantly depends on the architecture of the deep-learning

models. Similarly, the models trained using DL4 have better capability to predict sentiment as compared to other deep-learning techniques.

D. SMOTE

In this study, we have used two different variants of SMOTE techniques to handle the class imbalance nature of data. We have used box-plot and descriptive statistics of performance parameters to find the impact of data sampling techniques on sentiment analysis for software engineering comments. Figure 7 presents a visual representation of the predictive capability of models trained on a balanced dataset versus models learned on an imbalanced dataset. Table VII shows the descriptive statistics in terms of min, max, Mean, Median, Q1, and Q3 for models trained on sampled data and original data. From Figure 7, and Table VI, it can be seen that the models trained on sampled data have better performance as compared to the original data. The trained prediction models on sampled data have 0.76 average AUC, 0.88 average recall, 0.78 average precision, and 76 average accuracies. While, models trained on original data have 0.73 average AUC, 0.81 average recall, 0.82 average precision, and 81.51 average accuracy.

In this paper, the Friedman test with a significance level of 0.05 and two degrees of freedom has been considered to find the significant impact of sampling techniques on model performance. Table IV shows the mean ranks using the Friedman test for SMOTE, BLSTM, and original data. The smaller value of P represents that the models trained on sampled data have significant improvement in performance as compared to the original data. Similarly, the models trained on SMOTE sampled data have better capability to predict sentiments as compared to BLSTM and original data.

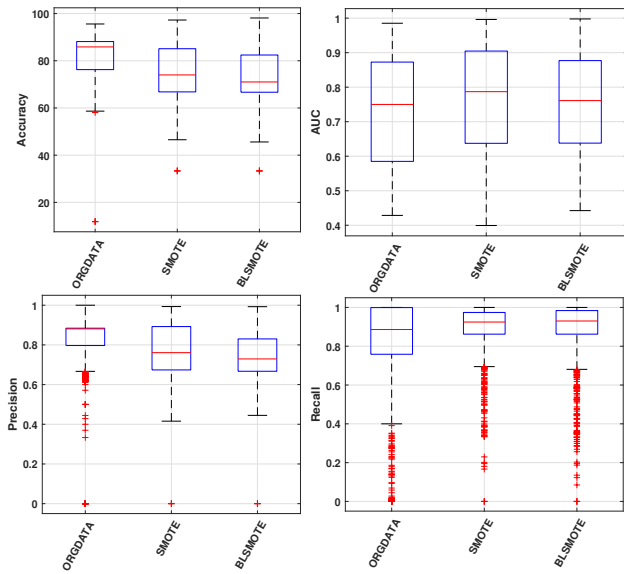


Fig. 7: Performance Box-Plot Diagram: Performance of Original Data and SMOTE

TABLE VII: Descriptive Statistics: OD and SMOTE

	Min	Max	Mean	Q2	Q1	Q3
Accuracy						
ORGDATA	11.87	95.57	81.51	85.86	76.24	88.13
SMOTE	33.33	97.25	76.00	73.99	66.81	85.09
BLSMOTE	33.33	98.11	74.16	71.01	66.67	82.43
Precision						
ORGDATA	0.00	1.00	0.82	0.88	0.80	0.88
SMOTE	0.00	0.99	0.78	0.76	0.67	0.89
BLSMOTE	0.00	0.99	0.75	0.73	0.67	0.83
Recall						
ORGDATA	0.00	1.00	0.81	0.89	0.76	1.00
SMOTE	0.00	1.00	0.88	0.92	0.86	0.97
BLSMOTE	0.00	1.00	0.87	0.93	0.86	0.98
AUC						
ORGDATA	0.43	0.99	0.73	0.75	0.59	0.87
SMOTE	0.40	1.00	0.76	0.79	0.64	0.90
BLSMOTE	0.44	1.00	0.75	0.76	0.64	0.88

VII. CONCLUSION

Sentiment analysis prediction models for software engineers help in various engineering tasks like analyzing developers' sentiments, evaluating app reviews, users' sentiments of software products, etc. The work presented in this paper is a successful effort in the direction of development of software sentiment models by using different variants of embedding techniques, different methods to find important features, different methods to handle the imbalanced nature of the dataset, and finally, different variants of deep-learning for model development. The performance of the developed models is computed and compared using accuracy, precision, AUC, and recall. We have also applied the Friedman test to statistically examine the performance of models developed using a different combination of features. The major findings

are summarized as follows:

- The high value of AUC for the trained models confirms the capability of the models to predict sentiment based on text comments.
- The use of sampling techniques like SMOTE and BLSMOTE significantly helps in improving the performance of software sentiment prediction models.
- The models trained by using selected sets of features using ANOVA achieved better performance as compared to other techniques.
- The deep learning with one dropout layer and two hidden layers achieved better performance as compared to other combinations.

VIII. ACKNOWLEDGEMENTS

This research is funded by TestAIng Solutions Pvt. Ltd.

REFERENCES

- [1] B. Lin, F. Zampetti, G. Bavota, M. Di Penta, M. Lanza, and R. Oliveto, "Sentiment analysis for software engineering: How far can we go?" in *Proceedings of the 40th international conference on software engineering*, 2018, pp. 94–104.
- [2] E. Biswas, K. Vijay-Shanker, and L. Pollock, "Exploring word embedding techniques to improve sentiment analysis of software engineering texts," in *2019 IEEE/ACM 16th International Conference on Mining Software Repositories (MSR)*. IEEE, 2019, pp. 68–78.
- [3] M. R. Islam and M. F. Zibran, "Leveraging automated sentiment analysis in software engineering," in *2017 IEEE/ACM 14th International Conference on Mining Software Repositories (MSR)*. IEEE, 2017, pp. 203–214.
- [4] L. Kumar, S. Misra, and S. K. Rath, "An empirical analysis of the effectiveness of software metrics and fault prediction model for identifying faulty classes," *Computer Standards & Interfaces*, vol. 53, pp. 1–32, 2017.
- [5] R. Jindal, R. Malhotra, and A. Jain, "Software defect prediction using neural networks," in *Proceedings of 3rd International Conference on Reliability, Infocom Technologies and Optimization*. IEEE, 2014, pp. 1–6.
- [6] G. I. P. Sari and D. O. Siahaan, "An attribute selection for severity level determination according to the support vector machine classification result," in *Proceedings of the 1st international conference on information systems for business competitiveness (ICISBC)*, 2011.
- [7] R. Malhotra and M. Khanna, "A text mining framework for analyzing change impact and maintenance effort of software bug reports," *International Journal of Information Retrieval Research (IJIRR)*, vol. 12, no. 1, pp. 1–18, 2022.
- [8] R. Malhotra and J. Jain, "Predicting defects in imbalanced data using resampling methods: an empirical investigation," *PeerJ Computer Science*, vol. 8, p. e573, 2022.
- [9] L. Kumar, M. Kumar, L. B. Murthy, S. Misra, V. Kocher, and S. Padmanabhuni, "An empirical study on application of word embedding techniques for prediction of software defect severity level," in *2021 16th Conference on Computer Science and Intelligence Systems (FedCSIS)*. IEEE, 2021, pp. 477–484.
- [10] R. Panigrahi, L. Kumar, and S. K. Kuanar, "An empirical study to investigate different smote data sampling techniques for improving software refactoring prediction," in *International Conference on Neural Information Processing*. Springer, 2020, pp. 23–31.
- [11] L. Kumar, S. K. Sripada, A. Sureka, and S. K. Rath, "Effective fault prediction model developed using least square support vector machine (lssvm)," *Journal of Systems and Software*, vol. 137, pp. 686–712, 2018.

Scrum, Kanban or a Mix of Both? A Systematic Literature Review

Necmettin Ozkan

Information Technologies Research and Development Center
Kuveyt Turk Participation Bank
Kocaeli, Turkey
necmettin.ozkan@kuveytturk.com.tr

Tugba Gurgen Erdogan

Computer Engineering Dept.,
Hacettepe University,
Ankara, Turkey
tugba@cs.hacettepe.edu.tr

Sevval Bal

Information Systems Engineering Dept
Sakarya University,
Sakarya, Turkey
sevval.bal@ogr.sakarya.edu.tr

Mehmet Şahin Gök

Department of Business
Gebze Technical University
Kocaeli, Turkey
sahingok@gtu.edu.tr

Abstract—Among the Agile methods, Scrum and Kanban are widely used in software development and they are considered the two most effective ones influencing the direct results of projects. Despite the importance of knowing their relative strengths and advantages and integrating them to achieve better results than individual use, none of the secondary studies provide extensive knowledge on the topic. In this paper, we performed a Systematic Literature Review (SLR) study to investigate the characteristics of the empirical studies which involve Scrum and Kanban by comparing or integrating them. Our final set includes 38 studies posing primary information on the advantages of each method over another one, the properties including artifacts, roles, and events from Scrum and Kanban in combining them in a hybrid way, and the properties of transitions from one to another such as transition directions (Scrum to Kanban, Kanban to Scrum or Scrum/Kanban to Hybrid), transition years, and transition reasons. The outputs can be interesting for both industry and researchers. For example, nearly all of the transitioning organizations are moving from Scrum to Kanban or to hybrid method. Among the reasons for the transitions, the problems experienced with Scrum are remarkable. In comparison, Kanban stands out clearly in a positive way. Almost all of the teams combining both use flow instead of a sprint.

Index Terms—Agile, software development, systematic literature review, SLR, agility, project management.

I. INTRODUCTION

IT IS a fact that due to the evolutionary nature of software development, changing market needs, and evolving technology [1], software projects inevitably change in many aspects including requirements, circumstances, and stakeholders [2], which require agility in complex domains. Based on this natural need for agility, people have invented varying agile approaches and methods to meet the need to be compatible in the market, have shorter development cycles, fewer costs, and have the ability to move and change quickly [3, 4]. Among the agile methods, Scrum and Kanban are common in the software industry [5] and they are considered the two effective agile methods that handle and manage the progress of software development [4, 6, 7].

Deciding on a development approach is one of the critical factors influencing the direct results of projects [8]. There has already been a debate for years about which of these methods (Scrum and Kanban) are preferred [7, 9]. These cases call for a proper and deep understanding of possible

methods, recognizing their strengths and weaknesses, limitations, relative advantages compared to others, context constraints, and so on. For instance, Scrum has limitations directly affecting application results, such as lack of work visibility, local optimization, large-scale implementations, and changing task priorities [10, 11, 12]. Similarly, as all other methods do, Kanban has some problems and challenges as well [13, 14]. Considering the individual limitations and challenges of each method, there are also some views that blending more than one method will yield better results than individual use. For instance, there are views that the limitations in Scrum can be mitigated by using Kanban alongside Scrum and they can complement each other [9, 15, 16, 17].

Some literature review studies have revealed the state of Scrum and Kanban separately. Despite the importance of knowing Agile methods closely, comparing them with each other, and using them together as a possible next step, none of the secondary studies provide extensive knowledge on the topic of comparing and/or integrating Scrum and Kanban to structure information available in the literature and highlight the opportunities for further research and practice.

In this paper, we performed a comprehensive literature review study to investigate the characteristics of the empirical studies which involve Scrum and Kanban by comparing or integrating them. We used a systematic literature review (SLR) as a research methodology which is a common method to conduct a literature review study. We manually searched for the studies published until March 2022 in the electronic digital libraries of scientific literature listed in Table I, and reached and deeply investigated 38 sources that compared or used Scrum and Kanban together.

The remainder of this paper is organized as follows. Section 2 summarizes the related works. In Section 3, we describe the overview research design with research questions and selection process. Section 4 delivers the results of the literature review. In Section 5, we discuss our findings and observations and in Section 6, we deliver limitations of the study and propose suggestions for future work.

II. RELATED WORKS

There are plenty of studies reviewing the Agile methods, comparing them with their characteristics, strengths, weaknesses, similarities, and differences, providing criteria to choose them according to the context of development

including those covered in this study. Apart from those included in our study, there are some other non-empirical studies comparing Scrum and Kanban such as study [18] and books such as [19] (However, as we focus only on empirical studies in our study, we excluded such unempirical ones).

There are some secondary studies close to our scope and systematically review the literature. For instance, study [20] systematically reviews the literature to identify the studies providing practices in requirements specification incorporated into the Agile development methods and delivers a comparison among Scrum, Extreme Programming, and Lean in this regard. From the systematic literature review, they found a total of 12 relevant studies and eight variability and commonality practices between these three methods. Study [21] systematically reviews the literature in order to analyze the current trends of Kanban usage in software development and to identify obtained benefits and involved challenges. Similarly, study [13] conducts a systematic mapping of Kanban literature in software engineering between 2006 and 2016 resulting in 23 primary relevant papers. It provides benefits, challenges and recommended practices of Kanban applications in software development and state-of-the-art opportunities for Kanban research. Similarly, there are many similar studies on Scrum as well. However, we have not found any study that systematically reviews the literature related to comparing and/or combining Scrum and Kanban.

III. RESEARCH DESIGN

This research process has been undertaken as an SLR based on the guidelines proposed by Kitchenham et al. [22]. The following section delivers the method used to conduct this SLR.

The research process starts with defining research goals and questions. After defining search queries and searching in the major digital libraries, we gathered 391 potentially relevant publications. For scanning the retrieved studies, we developed and applied inclusion/exclusion criteria and obtained a final pool of 38 sources. After extracting the data from the sources, the results of SLR are analyzed and the findings are discussed. The remainder of the section concerns the research questions, publication selection process, data extraction and synthesis, quality assessment, and potential threats to validity.

A Research Questions

This study aims to review and classify studies that compare and/or combine Scrum and Kanban. Thus, we set the main goals related to our research 1) identify the studies which compare and/or combine Scrum and Kanban and 2) analyze and synthesize the studies' results. Based on our study's goals, we raise and investigate four main and seven sub-research questions (RQs):

RQ1. Contribution types and research methods:

RQ1.1. Contribution types: How many sources have represented a new software process model, method, metric, or process/workflow in the software development domain using Kanban and Scrum together?

RQ1.2. Research methods: What types of research methods were used in the empirical studies by the authors?

RQ2. Advantages of each method over another one:

RQ2.1. Benefits of Kanban against Scrum: What benefits (quantitative or qualitative) have been reported as a result of applying Kanban against Scrum?

RQ2.2. Benefits of Scrum against Kanban: What benefits (quantitative or qualitative) have been reported as a result of applying Scrum against Kanban?

RQ3. What properties are used from Scrum and Kanban in combining them? What artifacts, roles, and events are combined and customized while combining. We aim to reveal some patterns through the consolidated list of the elements integrated in combinations of Scrum and Kanban.

RQ4. What are properties of the transitions from one method to another?

RQ4.1. What are transitions' directions (From Scrum to Kanban, Kanban to Scrum or Scrum/Kanban to Hybrid)

RQ4.2. Which years did transitions happened?

RQ4.3. What are the transitions' reasons?

B Publication Selection Process

The search process was a manual search of peer-reviewed studies in well-known digital libraries without any specific filter in the year range. Based on the scope of this study, the search string including "Kanban and Scrum" as "intervention" was developed by following the SLR protocol [22, 23]. We did not add "population" related keyword in the string referring to the application area which is software in our study to access the largest possible set, rather, this search was done manually by the authors in the papers. Regarding the search location, we anticipated and were satisfied with the effectiveness of searching in meta-data instead of the full text as the main aim of the paper is a comparison and combination of two methods in the software development domain in different contexts, then it is expected the authors locate the relevant terms in the papers' meta-data. Finally, the search process was done in March 2022 as shown in Table I.

TABLE I. NUMBER OF SOURCES RETRIEVED AND SELECTED BY SEARCH KEYWORDS

Search Library	Place	Search String	#Initially Retrieved	#Selected
Web of Science	Meta-data	Kanban AND Scrum	119	14
Wiley			2	1
Science Direct			6	3
Scopus			179	35
IEEE			62	14
Emerald			2	0
ACM DL			21	3
Total				
Total in Distinct			216	38

After defining the keywords and libraries, a pilot search was done by the first two authors to make sure the search process that would be applied is standard across the research. Based on the scope and context of our study, for the selection of papers, the following propositions of inclusion criteria (IC) and exclusion criteria (EC) in Table II were specified and applied to the papers, by focusing on empirical studies which

applied Scrum and Kanban to compare or integrate into the software development domain. As can be seen in the list, studies suggesting only the use of the Kanban board, which is similar to the Scrum board, and studies examining the use of Scrumban [16] which is also defined as a specific method, are excluded.

During the application of inclusion/exclusion criteria, the papers were examined through their titles and, where necessary, abstracts in order to identify whether they are within our scope. If the abstracts were not sufficient to decide to include or exclude the papers, then, scanning through the full texts of the papers was done to identify potentially relevant ones. After the first two authors of this study identified their selected papers, the paper lists obtained from these authors were then compared to each other until reaching a consensus between them. In this step, exclusions from and inclusions into the list were made, resulting in the final agreed-upon list. The whole search process was coordinated by the third author and reviewed by the fourth author to propose improvements if needed.

In the process, a total number of 391 peer-reviewed studies were returned from the search results as seen in Table I. This initial list included duplicate records since databases we covered perform meta-data indexing of publishing databases we included directly. After removing the duplicate records, the list included 216 distinct records. Out of these 216 papers, 12 papers were investigated only through their titles, 47 of them through their titles and abstracts, and 157 of them through their titles, abstracts, and full texts.

According to the ICs and ECs, 178 papers were excluded as seen in Table II. It is noted that five papers are not accessible by the authors, even though at the first glance, they seem relevant to our study scope. Specific to these five papers, a clear decision could not be made to include or exclude them because the title and summary information were not sufficient to decide and, then, they are excluded because there is no access to the full texts by the authors. We applied EC11 (papers not available in English) either by filtering via the library relevant features allowing eliminating non-English study beforehand or otherwise via manual investigations by the authors. After the manual investigation, the full texts of the five studies were not in English, and then, they were excluded. Consequently, 38 papers that compare and/or combine Scrum and Kanban within the scope of our study were identified. For the whole list, the spreadsheet containing the search results, together with inclusion and exclusion decisions is available online [24].

C. Data Extraction and Synthesis

A data collection form holding information was designed to record the collected information from the identified studies. The collected information ranges from general information about each study such as Author, Title, Year, Venue, Author Affiliation, and Author Country, as well as specific information to answer the research questions.

For the information that is subjective and open to evaluation, first, the relevant information was taken as it is from the relevant study and copied to the Excel file. Such information that is relatively difficult to quantify is such data from the parts about how the methods are combined and how they are superior to each other. Additionally, the parts of the studies including this particular information were highlighted in the original paper for any future references. For this type of data, it was aimed for researchers to have a bias as little as possi-

TABLE II. INCLUSION CRITERIA AND EXCLUSION CRITERIA

ID	Criterion	# of Eliminated Studies
IC1	Papers empirically validated and applied research methods like a case study, survey, action research, interview, simulation, literature review, experiment, focus group, pilot study, and statistical analysis	-
IC2	Papers comparing and/or combining Scrum and Kanban	-
IC3	Papers in software development domain	-
IC4	Conference, workshop, journal or book-chapter papers	-
EC1	Not empiric studies such as idea or opinion papers	26
EC2	Papers not related to comparing or combining Scrum and Kanban	84
EC3	Papers providing only a usage of Kanban board within Scrum	5
EC4	Papers providing only a usage ratio for Scrum and Kanban in any field	6
EC5	Papers providing only usage of Scrumban	1
EC6	Papers in other than the software development domain (for example healthcare, construction, supply chain, education, and teaching)	33
EC7	Papers published in non-peer-reviewed sources such as thesis, web pages, workshop proposals, tutorials, panels, proceeding information and, books.	13
EC8	Secondary studies such as systematic reviews or mapping studies	0
EC9	Papers not accessible by the authors	5
EC10	Duplicate studies	0
EC11	Articles not in English	5

ble and comment on the original data in this way and develop his/her comments gradually. In addition, four randomly selected studies were independently reviewed by the first three authors of this study and jointly evaluated until consensus was reached to ensure a common understanding and the data extraction step is applied in the agreed standard way. The rest of 34 remaining papers were allocated randomly to the first two researchers. The first and second researchers run separate sessions in order to extract the data that serves to answer the research questions by applying detailed and thorough examinations of the relevant studies. The third author coordinated the data extraction process. Consequently, a thorough reading of each of the 38 identified papers was performed to extract relevant information. Once data extraction was complete, the extracted data were closely synchronized and analyzed with the first two authors.

D. Quality assessment

The entire process relies on a search procedure that calls for explicit criteria to validate the quality of the selected candidate papers by ensuring each candidate paper is of adequate standard [22]. Accordingly, a custom quality-assessment-criteria-list and item descriptions were established as shown in Table III.

Each paper has been then assessed against this given set of questions by the assigned authors. A manual inspection was done through the full text investigation carried out to identify

TABLE III. CRITERIA FOR QUALITY ASSESSMENT

Criteria- Statement	Descriptions
QA1- Are contributions of method clear?	Clarity and robustness of the method applied in the study is satisfactory
QA2- Are outcomes as results clear?	Outcomes are clearly delivered and relevant to the method applied
QA3- Is discussion on results clear?	Discussion of the results is satisfactory and based on the results objectively. Validity threats are delivered.
QA4- Does paper have a citation?	Papers published at least two years ago have at least one citation.

each selected paper's quality assessment score. We set a score-weight based on the two values; Satisfactory (1) and Not Satisfactory (0). Accordingly, the evaluations of the papers have been made based on the predefined two values to set their scores yielding a score of four at maximum. It was decided that the studies with a score below two points would be eliminated. After applying the determined quality criteria, the quality scores of each study are as in the Excel Spreadsheet "Demographics & QA" available online at the previously shared Excel. As seen, there existed no studies assigned the lower than the threshold score and no elimination regarding the quality assessment was done.

IV. RESULTS

We present the results and findings of this SLR study concerning RQ1-RQ4. All sources included in this study are listed in a publicly accessible repository available at the previously shared online sheet named "Demographics & QA".

According to the results, 63% (24/38) of the studies are conference papers, 34% are journal articles, and 3% are workshop papers. In terms of venues for the selected 38 papers, "International Conference on Agile Software Development (XP)" is at the top with five papers, followed by "Hawaii International Conference on System Sciences (HICSS)" and "Agile Conference" with three papers for each. Regarding the journal articles, "Journal of Software: Evolution and Process" is the top one with two papers. In terms of the authors' affiliation types, 55% (21/38) of the papers are from academia, 24% from industry, and 21% from industry and academia collaboration. Related to the authors' country distribution, the USA is at the top with nine authors, followed by Norway, New Zealand, Brazil, the UK and Italy with three papers for each, among twenty-two different countries

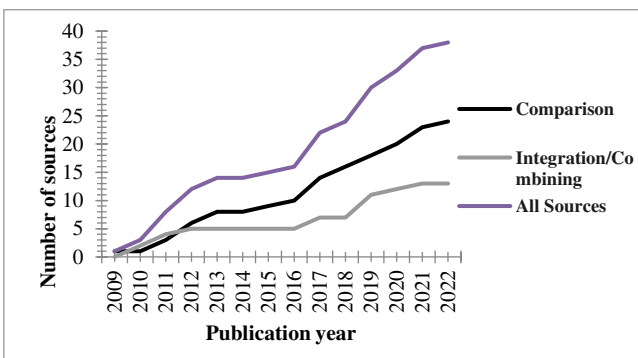


Fig. 1: Number of sources per year

in total. The contexts of the studies include software development (not specified) with seven papers at the top, software maintenance with three papers, and web-based content management system with two papers respectively. Other contexts include finance and insurance web services, automotive production software, content management system, library information system, data science, software project management, cloud-based software development, video game development, mobile software development and so on, among others. Figure 1 shows the cumulative number of sources per year by considering type (comparing vs. integrating).

Regarding RQ1.1 (Contribution types), Figure 2 depicts the contribution types of the primary sources. As shown, 22 studies do not provide any new model, method, tool, metric or process, as they rather mainly deliver a comparison between the methods or apply them. We see a considerable amount of new model proposals.

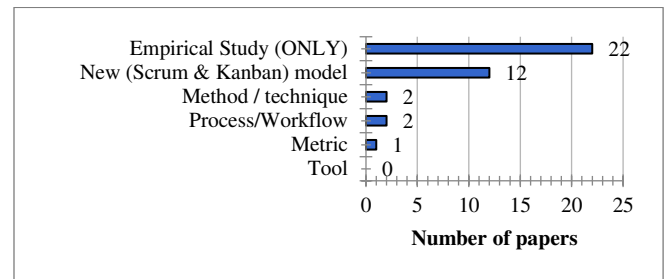


Fig. 2: Contribution types

In terms of the research methods used in the empirical studies by the authors (regarding RQ1.2), as seen in Figure 3, the majority of the studies use a case study method, followed by a survey, interview, and simulation.

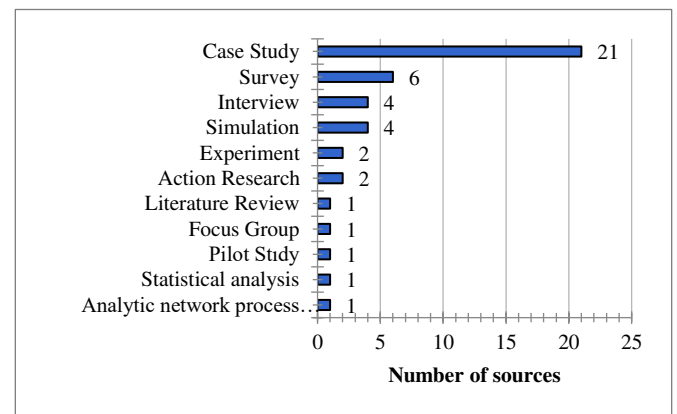


Fig. 3: Research methods

When it comes to RQ2.1 (Benefits of Kanban against Scrum), Table IV delivers the results. Similarly, Table V depicts results for Scrum Benefits against Kanban (RQ2.2), ordered by "Number of sources". We have added the domain information of each source by using abbreviations; 3D Animation Production as 3D, Agile Application (General) as Agile G., Agile Testing as Test, Automotive as Auto, Broadcasting Software Development as Broadcast, Cloud-based Software Development as Cloud, Content Management System as Content, Data Science Project Management as Data, Finance and Insurance Web Services as Finance,

TABLE IV: KANBAN BENEFITS AGAINST SCRUM

Category	Kanban Benefits against Scrum	Domain	Number of sources - Sources
Process	Flexible and adaptive	Mob, Web, Lrn, Lib, Agile, Game	7- [9], [30], [33], [36], [37], [4], [43]
	Easy for transition and use	Data, Lrn, Agile, Soft	7- [9], [35], [36], [4], [42], [45], [47]
	Efficiency	Soft, Broadcast, Cloud, UX	4- [48], [57], [58], [14]
	Focus on work	Lrn, Soft, Maintenance, Broadcast	4- [36], [54], [55], [57]
	Quality	Agile, Game, Soft P.	4- [9], [4], [43], [53]
	Delivery time	Content, Game	3- [9], [38], [43]
	Visibility	Maintenance, Broadcast, Soft P.	3- [7], [55], [57]
	Performance	Agile, Soft, Maintenance	3- [25], [4], [46]
	Controlling the flow	Web, Cloud	2- [40], [58]
	Delivering value	Lib, Soft	2- [37], [54]
	Project schedule management	Soft P.	2- [7], [53]
	Project resources management	Soft P.	1- [53]
	Project risk management	Soft P.	1- [53]
	Consistency in project management	Soft P.	1- [7]
	Being reliable	Agile	1- [4]
	Better for development with no certain deadline	Lrn	1- [36]
	Better for small and not complex feature development	-	1- [9]
	Closer contacts with users	UX	1- [14]
	Cost-effective	-	1- [9]
	Eliminating excessive and unnecessary meetings	Mob	1- [30]
Less rework	Lib	1- [37]	
Getting fast feedbacks	-	1- [9]	
Traceability	Maintenance	1- [55]	
People	Teamwork	Maintenance, Data	2- [35], [55]
	Collective understanding	Maintenance, Broadcast	2- [55], [57]
	Respect for people and their current states	Soft	1- [42]
	Satisfaction of individual team members	Data	1- [35]
	Less stress	Mob	1- [27]

TABLE V: SCRUM BENEFITS AGAINST KANBAN

Category	Scrum Benefits over Kanban	Domain	Number of sources - Sources
Process	Path clarity, being well-defined	Soft, Video	3- [42], [43], [47]
	Better for development with batched and a block of works	Agile	2- [9], [4]
	Delivery time	Maintenance, Soft	2- [25], [47]
	Better for bigger teams	UX	1- [14]
	Better for development with certain deadline	Lrn	1- [36]
	Better for development within high uncertainty environments	Agile	1- [4]
	Better for the testing process	Test	1- [31]
	Project cost management	Soft	1- [53]
	Detailed tracking and overview of projects	Lrn	1- [36]
	Predictability	Soft	1- [54]
	Specification of customer requirements	Soft	1- [47]
	Easy to manage	Soft	1- [54]
People	Teamwork	Web, Soft	3- [33], [42], [47]
	Empowering	Mob	1- [27]
	Communication	Video	1- [43]

Learning Management System as Lrn, Library Information System as Lib, Mobile Software Development as Mob, Open Source as Open, Software Development (General) as Soft, Software Development Project Management as Soft P., Software Maintenance as Maintenance, System Admin as System, Telecommunication Software Development as Telco, User Experience as UX, Video Game Development as Game, and Web-based Content Management system as Web. In column “Category”, the classified information of each item including process, organization and human dimension is given. The “Organization” category is about making decisions

related to software engineering and development in a business context and aligning software technical decisions with the business goals of the organization [13].

Kanban is regarded as more flexible and adaptive than Scrum without strict time-boxes, roles, rules and sprint constraints. Kanban teams feel more powerful to address their internal processes and respond quickly with its continuous flow management allowing constant re-planning to uncertainty and frequent changes and faster responses. They can deliver results earlier with frequent releases and division of the work in very small-time chunks [37], [48]. Kanban is

also simpler and paves the way to an easier transition with less cost, time, chaos and turbulence. It does not touch titles or positions of people, requiring a lower “patience point” [42]. Thanks to its more “sequential” nature and WIP limits, Kanban seems to be more efficient in distributed context [58]. Using Kanban over its precursor Scrum practices brings advantages in terms of efficiency and focus with WIP limits, stopping context-switching, providing granularity of visualization, gaining a collective understanding of the whole process, less paperwork, and reducing batch size through the pipeline [57], [14].

As reported quantitatively and/or qualitatively by some studies, Kanban provides better quality, project schedule, risk, and resource management, reliability, lead time results, performance, visibility, traceability, teamwork, the satisfaction of individual team members, regular feedback faster, and closer contacts with users. It does not require haste like in Scrum and, therefore, resulted in better quality [4] and less stress [27]. Kanban works better within certain contexts like development with no certain deadline and small and not complex feature development. It allows for identifying reworks at early stages [37]. It has been found that Scrum supports collectivist teamwork, more empowered team members, shared goals and intense communication across the team. Although Scrum has a rigid and routine structure, it provides a path clarity and easiness to manage the processes thanks to its being a well-defined method. According to some results, Scrum provides a better cycle time. Scrum also seems to be more suitable for bigger teams, batched works and developments within high uncertainty environments like for new product development and within certain deadlines. Some of the results quantitatively put forward that it is better than Kanban in terms of project cost management and agile testing processes. Scrum provides better predictability and detailed tracking and overview of projects.

Regarding RQ3 (properties of the proposed combining models), Table VI depicts the results by providing what properties are used from Scrum and Kanban in combining them (M stands for Method). It shows in particular what artifacts, roles, and events are combined and customized while combining Scrum and Kanban. In the Scrum and Kanban integration, numerically speaking, the most used elements are PO (Product Owner), Daily, Sprint Review, Sprint Retrospective, Scrum Team, User Story, Definition of Done, and Sizing from Scrum, and, Pull system, Continuous flow, WIP, and Kanban boards from Kanban. Besides, there are some custom-made elements in their integrations. Notable ones can be listed as follows: simultaneously used number and size-based WIP, iterative movement of the items on the flow, work/hypothesis/experiment/release-based iterations, calendar-based regular or on-demand ceremonies, daily planning, team formations that break the “standard” cross and self-organizing team structures including floating teams, sub-groups/roles and supervisory authority, same/similar-size work items, and product owner teams.

TABLE VI: ELEMENTS USED IN COMBINING

M	Element/Source	Number of sources - Sources
Scrum	PO	11 - [26], [28], [29], [32], [33], [34], [39]-Case1, [40], [41], [49], [56]
	Daily	9 - [26], [28], [33], [38], [41], [44], [49], [51], [52]

	Scrum Team	7 - [26], [28], [39]-Case1, [49], [51], [52], [56]
	Product Backlog	6 - [28], [32], [34], [44], [49], [56]
	User Story	5 - [26], [28], [32], [38], [51]
	Definition of Done	5 - [28], [34], [38], [44], [51]
	Sizing (T-Shirt size, effort-based, planning poker etc.)	5 - [39]-Case2, [40], [49], [51], [56]
	Sprint Review	5 - [28], [34], [38], [44], [49]
	Sprint Retrospective	5 - [28], [34], [38], [44], [49]
	Grooming	3 - [34], [49], [56]
	SM (Scrum Master)	3 - [28], [33], [49]
	Definition of Ready	2 - [28], [51]
	Sprint Planning	1 - [26]
	Minimal Marketable Feature-Sets / Minimum Viable Product	1 - [41]
	Sprint	1 - [26]
	Kanban	Kanban boards
Work in progress (WIP)		10 - [28], [33], [34], [39]-Case1, [40], [44], [49], [51], [52], [56]
Pull system		7 - [34], [38], [40], [41], [44], [51], [56]
Continuous flow		5 - [33], [39]-Case1, [40], [41], [56]
Metrics		3 - [39]-Case2, [40], [44]
Value-stream/chain		2 - [28], [41]
Service level agreement/expectation		2 - [28], [39]-Case2
Size of task is not limited		1 - [56]
Replenishment		1 - [52]
Kanban coach		1 - [52]
Custom	Calendar-based regular ceremonies	4 - [40], [49], [52], [56]
	Work/Experiment/Hypothesis/Release-based iteration planning	4 - [29], [32], [49], [51]
	Back and forward movement of the items on the flow	3 - [26], [40], [56]
	Product Owner Team	3 - [44], [51], [52]
	Supervisory authority	2 - [29], [32]
	Same/similar-size work items	2 - [40], [41]
	WIP (number and size based)	1 - [26]
	Metrics: Cycle and Lead time integrated with story point (pseudo-velocity)	1 - [26]
	Buffer in capacity	1 - [26]
	Backlog clean-up actions	1 - [29]
	Not multi-functional teams	1 - [32]
	Ceremonies scheduled on demand	1 - [32]
	Sub-group/role-based board columns	1 - [32]
	Leader responsible for tasks administration on the board	1 - [32]
	'Wiki' to keep track of project's progress rather than the Scrum events	1 - [32]
	Daily Planning – a combination of Sprint Planning and Daily Scrum	1 - [34]
	A person to take care of processes and team development	1 - [34]
Review for finished items on the board	1 - [34]	

Floating teams (distribution of team members between different floating teams)	1 - [38]
Each feature owned by a developer	1 - [38]
Longer [than the maximum in Scrum] iterations	1 - [40]
Team Coaches	1 - [44]

Related to RQ4 (Properties of the transitions from one method to another), the transition directions are from Scrum to Hybrid (11/18), from Scrum to Kanban (5/18), from Waterfall to Hybrid (1), and from Custom to Hybrid (1/18). It can be seen that most of the transitions are to the hybrid methods and predominantly from Scrum. Additionally, Figure 4 and Table VII present transition years, and transition reasons respectively (the studies that explicitly express the transition year form the cumulative A curve, and after adding the five studies that do not explicitly express the transition year from the cumulative B curve when the year of publication is accepted as the transition year).

For the transition reasons, regarding the lack of flexibility and predictability during the fixed sprints and timeboxes, the studies report that sprints are not adequate for frequent, constant, and unexpected changes and (especially external) dependencies in hectic and unstable environments that require quick responses, frequent re-planning, re-prioritization, and a vast amount of mid-iteration updates for dynamism or resulting in non-predictive sprints. These issues make Scrum unsuitable for maintenance teams in which the dynamism is high and estimation is time-consuming and often incorrect. Under these circumstances, it becomes unclear how much teams would deliver in a sprint [40].

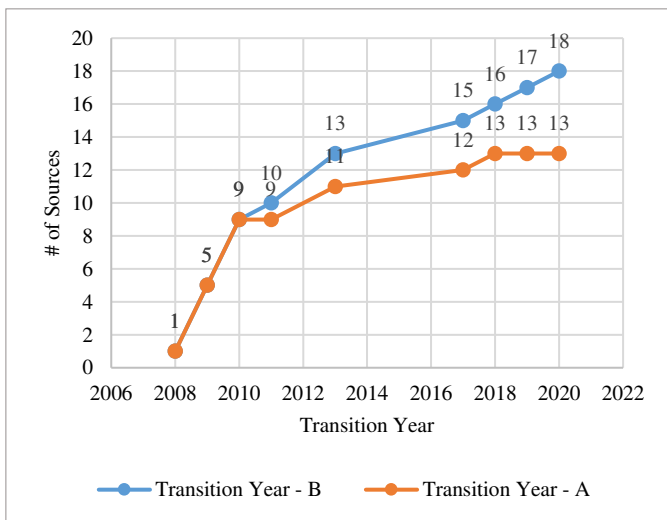


Fig. 5: Transition Year

The combination of inaccurate estimates and timeboxes leads to non-sensing estimation activities and, thus, waste and reduced productivity [38], [46]. Some studies indicate that Scrum causes constant strain and quality issues such as increased technical debt, decreased system maintainability, untested implementations, and unmet production maintenance just to meet business time targets leaving little time to “others” including technical and administrative tasks with “no business value” or “little urgency”. Such tasks mostly remain in the background of the business. Study [38] states that developers did not pay enough attention to creating good architectural

designs, rather mainly focused on the short-term goals of the sprints. Short-term sprints also decrease visibility beyond sprints. Some teams regard Scrum as difficult to accept with its revolutionary transformation change proposition [45], “aggressive”, “materialized” and “rigid” structure [42], [46].

TABLE VII. TRANSITION REASONS

Category	Transition reason	Domain	Number of sources
Scrum			
Process	Lack of flexibility and predictability during sprints for frequent/unexpected changes	3D, Telco, Mob, System, Web, Maintenance, Open, Broadcast	11- [28], [29], [32], [33], [34], [40], [44], [45], [55], [56], [57]
	Quality decrease	Telco, Content, Web, Auto, Soft	5- [29], [38], [39], [40], [46]
	Unsuitable for maintenance	System, Maintenance, Soft	4- [34], [44], [46], [55]
	Estimation is time-consuming, non-sensing, or causing delays	Web, Content, Soft	3- [38], [40], [46]
	Scrum ceremonies cost	Mob, Finance	2- [32], [41]
	Lack of work visibility	Maintenance, Finance	2- [41], [55]
	Focusing on short-term goals of sprints	Content	1 - [38]
	Challenges in Sprint-based delivery	Web	1 - [40]
	Increased length of feedback loops	Finance	1 - [41]
	Being rigid	Soft	1 - [46]
	Failed sprints	Soft	1 - [54]
	Reduced productivity	Soft	1 - [46]
	Work items rarely correlated	Web	1 - [40]
	Lack of overall status beyond sprints	3D	1 - [28]
	External dependency	Open	1 - [56]
Ineffective Scrum events	Content	1 - [38]	
Waste	Soft	1 - [46]	
Organization	Challenges in large scale projects	Mob, Content, Soft	3- [32], [38], [46]
	Fragmentation	Web, Content, Telco	3- [29], [38], [40]
	Revolutionary change	Telco	1 - [45]
	Lack of co-location	Web	1 - [40]
	Lack of communication and collaboration	Maintenance	1 - [55]
People	Challenges in having a common language with business	Content	1 - [38]
	Need for deep specialty in teams	Web, Open	2- [40], [56]
	Constant strain	Telco	1 - [29]
	Challenges in being self-organizing teams	Telco	1 - [29]
	Challenges in having cross-functional teams	Web	1 - [40]
Lack of team lead resulting in a stagnation of team development	Telco	1 - [29]	
Kanban			
Process	Lack of control	System	1 - [34]
	Lack of deadlines	System	1 - [34]
People	Not having a feel progressing	System	1 - [34]

This strict position of Scrum also seems problematic when a lack of co-location is present [40]. We have seen some scaling challenges with Scrum including issues in the distributed scenarios, breaking large items down into smaller ones to be squeezed into one sprint, lack of synchronization, and fragmentation caused by isolated and localized Scrum teams. Among other items, specialization in certain parts of systems, dependency on deep specialists, and rarely correlated work items to bunch for/in a sprint are counted. Scrum ceremonies can cause rework, high cost, increasing length of feedback loops, lack of end-to-end work visibility, and lack of visibility within sprints. They, then, are done more and more seldom, shorter and shorter in their duration to finally “die out” [38] or end up with failed sprints [54]. Scrum teams may result in stagnation and having dominant team members and group formations within the team when team members are unchanged [29] or some other issues when teams have a number of part-time resources with varying availability [40]. On the other hand, in Kanban, with the lack of boundaries, teams are prone to drive into “guerilla-style” working and feel a lack of deadlines and no actual sense of progress.

V. DISCUSSION

The fact that the majority of sources were produced by industry-oriented authors, especially as experience and case study papers, indicates that the subject of the study can be considered mainly as a practical-led area and a need by the sector. This shows that the industry is ahead of academia in this regard. The data about a cumulative number of sources per year indicates that the first studies start just after the years when the first Kanban documents were published, the two methods are mostly compared in the first years, and the sources on the integration gained speed later on but have slowed down in recent years, which needs a further investigation. In general, after 2016, a noticeable acceleration in the number of sources draws attention.

Static Entities: It seems that there are some (almost) static entities in Scrum including sprint, time-boxes, and team formations that are found problematic in terms of providing flexibility and agility. Scrum plans and starts the sprint with a prediction for the future, but where the external environment change is high, the sprint predictions become incapable. This indicates that the sprint structure provides low agility in a high change environment. With this level of inflexibility, it seems that one of the areas where Scrum is most incapable is maintenance work. The main reason for Scrum's inadequacy in maintenance works is that these environments have a higher level of change than other environments. With a sprint-based maneuverability, Scrum works more comfortably in environments with low/medium uncertainty. Where the uncertainty is high, the estimations made to fit into this artificial and such static sprints become more meaningless and seen as a waste. All these are mainly why people regard Kanban and continuous flow along with its metrics designed for dynamic work management in the hybrid solutions as more flexible, adaptive, and powerful compared to Scrum.

Packaging work items: Sprint also focuses on packaging work items with a development-oriented perspective so-called the artificial world of development, rather than the real needs of the business world, so-called business-oriented development. In this sense, sprint is also (for example) a method of fragmentation. Even if things are not in a nature to be broken up or to be joined, they are compelled to do so.

Deadlines: Sprint includes deadlines, even measured in seconds; the end of a sprint. With a production-based logic, Scrum locates people as production muscles and expects them to comply with the strict lines, stressing and tiring them out. Estimates for the works to be done within the deadlines continue to be made in Scrum with the classical method logic; The main thing changing is the unit (man/day versus story point). With these concrete “lines” and the “glorification” of business orientation, parts that look inward (to technical sides and systems) may also stay in the background. To assign a static end rather than an end according to the nature of the works creates a meaningless and dangerous dichotomy between the nature of the work and these static ends. It is dangerous because the developers may not always choose the right way (e.g. even if things are not “on the way”, do tests properly). This dilemma paves the way for the sacrifice of more “abstract” concepts such as quality. In order to solve this problem, it is seen that some hybrid solutions are adopting business-work/experiment/hypothesis-based iteration planning solutions, not clock-time-based ones.

Push System: In Scrum, the flow is partially “push” based, where tasks would move to the next step as soon as they completed the previous step [59]. Also, Sprint Planning works for how much work to push (into the sprint) [12]. Scrum further “pushes” the teams to produce shippable products at the end of the sprint [12]. This push and time-box approach in Scrum [39] along with artificial hot deadlines of the sprints may lead to more stress, pressure, and fatigue on the teams and enforce them to compromise on quality. Kanban WIP limits aim to reach a near real-time balance between demands and capacity in a way free of stress for developers. Using a persistent, balanced, and thus, stress-free flow in Kanban facilitates achieving a more sustainable and reliable flow [59] that positively affects the final product and its quality. The pull system applying WIP limits stands as a choice of many hybrid solutions reported in our study.

Inside of the Sprint: In Scrum, the inside of the sprint is like a single station of which WIP is sprint-based. It makes the default, fundamental and granular level of work management level sprint-based. Thus, work management inside a sprint is less trackable. For this reason, congestion and bottlenecks may also occur in certain parts of a sprint. Kanban offers a more effective control mechanism with WIP limits that seem to be widely integrated into the hybrid solutions over controlling capacity and scope of work under high variability in Scrum [40]. In Kanban, each workstation is designed separately and traceability and visibility are provided to all teams and stakeholders through a continuous workstation-based flow on the Kanban board (even not reset at the end of the sprint). Kanban board can be considered as an advanced version of Scrum board in this sense and it can be seen that it is widely preferred among the hybrid solutions.

Scaling, with Isolation: The original Scrum setup is team and sprint-based, posing challenges in large-scale projects. This issue is manifested in our results as fragmentation, challenges in large scale projects, focusing on short-term goals of sprints, lack of overall status beyond sprints, and lack of communication and collaboration issues as shown in our “Transition Reasons” list. On the other hand, Kanban aims to gather the teams around the value streams for customers and address the whole organization by adopting system thinking and Fit-For-Purpose principles [60] on the horizontal axis that starts with the customer and ends with the customer delivery.

The indication of this aim in the hybrid solutions is the use of value-stream/chain.

A sprint backlog is owned by one specific team unlike a Kanban board shared by multiple teams [14]. Similarly, when sprint (specific to a particular team), *self-* organizing and cross-functional team promises come together, an encapsulation, division, and, to an extent, isolation issues occur for the teams. Due to its nature and instinct, sprint is a kind of planning within a particular team, while inter-team planning should be done according to emergent principles and globally. Despite the scaling models, doing this alignment within the teams at longer intervals emerges as a problem in Scrum.

Moreover, In Scrum, breaking the work down into small items regarding the time constraint and reductionist slicing of work can disrupt the holistic view and loss of track of item dependency [14]. Thus, sprint breeds an instinct for a short-term approach. Kanban supports better project management results and team building around common and collective understanding beyond sprints. Work items can be delivered earlier, as Kanban implements WIP in the entire flow and items can behave independently, eliminating [unnecessary] dependency within work items (bundling items in a sprint is a kind of dependency).

Necessities(!): With its flexible structure, Kanban allows teams to be more self-organizing. Thus, activities that are not necessary and done at specific times just to comply with the "necessities" of the framework can be avoided (The calendar-based, not sprint-based events, in the hybrid solutions are a good example of this). Moreover, teams can develop actions in a more sensitive, lively and instantaneous (synchronous) manner in addition to the a-synchronous actions offered by the framework. For example, teams should not wait for the end of the sprint for feedbacks or for the beginning of the sprint for new work to take in.

Scrum provides a form with its "rigid" structure for cultures that need order that is relatively missing in Kanban. In contrast, each "game rule" in Scrum is hand-binding, and Scrum's predictable, rigid, artificial boundaries promise an average-level-performing goal and may hinder average teams from performing higher. "Increasing the length of feedback loop" is an example of this; Because it recommends that feedback is *mainly* managed through an artificial structure, not emergently. Generally speaking, Scrum's addiction to asynchronous communication can be a challenging factor where synchronous communication is the remedy. Scrum events are a sort of prediction. Their ability to manage a different flow is low. On the other hand, since Kanban does not come with a relatively rigid form, it seems more challenging for it to take a shape. Moreover, Kanban teams are expected to be internally motivated. The driving force is internal, not external [as in Scrum]. In teams that do not have this intrinsic strength, therefore, applying Kanban can make the situation worse. Even for motivated teams, being in a homogeneous and flat flow in Kanban may not support the feeling of progress. Against the lack of control, not having a feeling of progress, and lack of deadline matters in Kanban, Scrum's sprint structure comes with solutions.

Transition: The fact that Kanban is closer to Lean as root than Scrum seems to have made it simpler. Scrum, rather, is like a luxury framework with its add-ons at a high cost. This luxury also applies to the transition to it. Even at the entry

level, Scrum requires a threshold to endure, and teams must be accompanied until they pass this threshold in a healthy way. Because the transition steps are bulk, teams suddenly find a more tough challenge and have to deal with it even in the first place. Radical transitions to this rigid and costly structure seem to threaten and wear out the teams.

In Kanban where progress rather than transition is essential, the transformation is easier. Kanban focuses on the flow (process) rather than providing agility through teams' (re)organizations for transitions. Kanban does not also require any strictly pre-defined roles or processes adopted in a revolutionary way like in Scrum, rather recommends an iterative and incremental (shortly agile) way of implementation and an easy way for transition and use. It does not have a special preference and compulsion regarding the setup of the teams. In this sense, with its principle "start with what you do now" and "initially, respect current processes, roles, responsibilities and job titles" [59], it can align and coordinate existing teams at upper levels. However, as reported by study [13], it similarly seems from our results that current studies covered in our work are mostly restricted to process and people levels and have not yet been scaled to the organization level including such as portfolio project levels or being used as a tool for decision-making by management.

Miscellaneous (Unclassified): Kanban does not deal with the types of works in the flow, they are regarded as homogeneous. Scrum, on the other hand, considers the works as complex by default and allows to handle such works according to their nature. Scrum, thus, seems to be more advantageous than Kanban in complex projects.

Scrum is better at cycle time while Kanban is better at lead time. Scrum encourages more collaboration within the team. While Scrum is better in intra-team communication, it seems problematic in inter-team communication. Scrum teams do not always go on the "happy path", the other side of the coin comes to the surface and it is seen that the problems between people (experienced in hierarchical structures) are present to them as well.

Depending on the situation, specialties still seem necessary by considering a balance to strike between specialization and generalization. Although some issues with the PO role generating an extra layer and dependency have been witnessed, the hybrid solutions including the PO role have been commonly observed. It means there seems to be a need for a mediator, the PO role, which acts as a bridge between the customers and the development teams. Having the PO in the process may also ensure that a representative of the customer is involved. The PO can also bring simplicity to the structure.

Both Scrum and Kanban seem to be good in delivery time, tracking and overview of projects, and teamwork. Where one is weak, the other one can be strong. These and similar situations still make these two methods preferable. However, the reported disadvantages of the particular methods (especially Scrum) open gates to integrating the most advantageous part of the methods and eliminating their disadvantages at the same time by hybridizing them.

Regarding the hybrid models, sprint is almost non-existent, instead, flow is extensively used. In relation to this, it is seen that the SM role is also positioned very few. Mostly, the roles and events are integrated from Scrum. One of the

reasons can be an endeavor to use the ready-roles of Scrum that are not defined in Kanban or not abandoning already-used Scrum roles after the transition to a hybrid one. We have seen that Daily is used as a common communication manner in the hybrid models. Nevertheless, it seems that some teams do not prefer to feed a rigid and relatively bloated framework, like Scrum, just for the sake of it. They tend to avoid the high cost of events of Scrum and (seldomly) want to make them when necessary or (mostly) on a calendar base. However, there is a tendency to do the events as routine (calendar/iteration/release) based rather than only as needed. Additionally, there is still a desire to predict the future by the teams, but they do not want to limit themselves to a deadline. In providing forecasting/estimations, there are certain studies that consider not only size but quantity and size criteria together. Some solutions break Scrum's rigid team structure and redesign it according to the needs. Similarly, the solution that converts Kanban's one-way flow to a two-way flow direction is also observed.

We have seen that Scrum has advantages in requirements management phases and many solutions are trying to address these phases with Scrum particles. Especially in this area, there is a preference for the product-based structuring that may be easier to manage with Scrum. In general, while Scrum seems good at defining the projects, Kanban stands out in the development part of it.

In our study results, it has been observed that Kanban provides an advantage over Scrum in general. The results of the study also exhibit a considerable percentage of transitions to hybrid models (from Scrum). Most organizations adopted Scrum before Kanban [24]. This may be one reason why the transitions are mostly from Scrum. On the other hand, studies [38, 46, 55] report that an increasing number of companies previously using Scrum is switching to Kanban due to the issues that Scrum bears.

Considering Kanban's linear and Scrum's deterministic approaches with the effect of the manufacturing sector to a certain extent, one of the recommendations of our work would be to use Scrum and Kanban with the "AND" operator, not "OR" as we have seen how they complement each other and how they produce better solutions together. We recommend providing varying hybrid models relevant to different contexts of organizations. When choosing a method, it is more appropriate to adopt it in a way that will be beneficial to the organizations themselves [14], rather than fully complying with that particular method; it is recommended to blend the methods according to the needs instead of blindly adhering to one method. For such an integration, it is reminded that Kanban is methodology independent; it can be applied to teams implementing Scrum, Extreme Programming, or Waterfall at the operational levels [39]. It is also proposed that articles that are going to be focusing on the integration of multiple Agile methods might be providing more insightful information, especially for the practice.

VI. LIMITATIONS AND FUTURE WORK

The procedures used in our study have limitations in several ways. More likely, we may have missed some relevant studies as we did not include all possible libraries. In particular, we have missed the studies published in non-peer-reviewed resources. Although the studies included and excluded were checked by other researchers, for most of the studies, a single researcher extracted the data from the

included studies. The values of the quality assessment criteria are somewhat subjective. Also, the primary studies' results are context dependent and their generalizability may be low and problematic when ignored.

We are planning a further study that will evaluate the combining models to grab their common patterns from different views and we will recommend our own new model(s) with an eclectic approach. The proposed models will be suited best for certain contexts and be selectable accordingly with guidance to know the cost, trade-offs and (dis)advantages of the selections. Before finalizing the proposed models, we will conduct evaluations with experts in the field of Agile Software Development to develop the model iteratively and incrementally.

REFERENCES

- [1] Nouredine, A. A., Meledath, D., & Samira, Y., "A Framework for Harnessing the Best of Both Worlds in Software Project Management: Agile and Traditional", Information Systems Education Conference, 2009.
- [2] Henderson, P., "Why large IT projects fail", *ACM Transactions on Programming Languages and Systems*, vol. 15, no.5, pp.795–825, 2006.
- [3] Conn, S., "A New Teaching Paradigm in Information Systems Education: An Investigation and Report on the Origins, Significance, and Efficacy of the Agile Development Movement", *Information Systems Education Journal*, vol. 2, no. 15, pp.3 – 18 2004.
- [4] Zayat, W., & Senvar, O., "Framework study for agile software development via scrum and Kanban", *International journal of innovation and technology management*, vol. 17, no.4, 2020.
- [5] Weflen, E., MacKenzie, C. A., & Rivero, I. V., "An influence diagram approach to automating lead time estimation in Agile Kanban project management", *Expert Systems with Applications*, 187, 2022.
- [6] Alaidaros, H., & Omar, M., "Software project management approaches for monitoring work-in-progress: A review", *Journal of Engineering and Applied Sciences*, vol. 12, no.15, pp. 3851-3857, 2017.
- [7] Lei, H., Ganjezadeh, F., Jayachandran, P. K., & Ozcan, P., "A statistical analysis of the effects of Scrum and Kanban on software development projects", *Robotics and Computer-Integrated Manufacturing*, vol. 43, pp.59-67, 2017.
- [8] Aurisch, R., Ahmed, M., & Barkat, A., "An outlook at Agile methodologies for the independent games developer", *International Journal of Computers and Applications*, vol. 43, no.8, pp. 812-818, 2021.
- [9] Alqudah, M., & Razali, R., "An empirical study of Scrumban formation based on the selection of scrum and Kanban practices", *Int. J. Adv. Sci. Eng. Inf. Technol*, vol.8, no.6, pp.2315-2322, 2018.
- [10] Tripathi, N., Rodriguez, P., Ahmad, M. O., and Oivo, M., "Scaling kanban for software development in a multisite organization: Challenges and potential solutions", *Agile Processes, in Software Engineering, and Extreme Programming*, pp. 178–190, 2015.
- [11] Rodriguez, P., Partanen, J., Kuvaja, P., and Oivo, M., "Combining lean thinking and agile methods for software development: a case study of a Finnish provider of wireless embedded systems detailed", *47th Hawaii International Conference on System Sciences (HICSS)*, pp. 4770–4779, IEEE, 2014.
- [12] Ozkan, N., & Gök, M. Ş., "How Scrum Inhibits Agility", *15th Turkish National Software Engineering Symposium (UYMS)* pp. 1-6, IEEE, 2021.
- [13] Ahmad, M. O., Dennehy, D., Conboy, K., & Oivo, M., "Kanban in software engineering: A systematic mapping study", *Journal of Systems and Software*, vo.137, pp.96-113, 2018.
- [14] Law, E. L. C., & Lárusdóttir, M. K., "Whose experience do we care about? Analysis of the fitness of scrum and kanban to user experience", *International Journal of Human-Computer Interaction*, vol.31, no.9, pp.584-602, 2015.
- [15] Banijamali, A., Dawadi, R., Ahmad, M. O., Similä, J., Oivo, M., & Liukkunen, K., "An empirical study on the impact of Scrumban on geographically distributed software development", *4th international*

- conference on model-driven engineering and software development (MODELSWARD), pp. 567-577, IEEE, 2016.
- [16] Ladas, Cc. *Scrumban-essays on kanban systems for lean software development*: Modus Cooperandi Press, 2009.
- [18] Alqudah, M., & Razali, R., "A comparison of scrum and Kanban for identifying their selection factors", 6th International Conference on Electrical Engineering and Informatics (ICEEI) pp. 1-6, IEEE, 2017.
- [19] Kniberg, H., & Skarin, M., *Kanban and Scrum-making the most of both*: Lulu. com, 2010.
- [20] Herdika, H. R., & Budiardjo, E. K., "Variability and commonality requirement specification on agile software development: Scrum, xp, lean, and kanban", 3rd International Conference on Computer and Informatics Engineering (IC2IE), pp.323-329. IEEE, 2020.
- [21] Ahmad, M. O., Markkula, J., & Oivo, M., "Kanban in software development: A systematic literature review", 39th Euromicro conference on software engineering and advanced applications, pp. 9-16, IEEE, 2013.
- [22] B. Kitchenham, O. P. Brereton, D. Budgen, M. Turner, J. Bailey, and S. Linkman, "Systematic literature reviews in software engineering--a systematic literature review," *Inf. Softw. Technol.*, vol. 51, no. 1, pp. 7-15, 2009.
- [23] B. Kitchenham and S. Charters, "Guidelines for performing Systematic Literature reviews in Software Engineering Version 2.3," *Engineering*, vol. 45, no. 4ve, p. 1051, 2007.
- [24] <https://tinyurl.com/2p89fvex>.
- [25] Anderson, D. J., Concas, G., Lunesu, M. I., Marchesi, M., & Zhang, H., "A comparative study of Scrum and Kanban approaches on a real case study using simulation", *International Conference on Agile Software Development*, pp. 123-137, Springer, Berlin, Heidelberg, 2012.
- [26] Polk, R. "Agile and Kanban in coordination", *Agile Conference*, pp. 263-268, IEEE, 2011.
- [27] Laanti, M., "Agile and Wellbeing--Stress, Empowerment, and Performance in Scrum and Kanban Teams, 46th Hawaii International Conference on System Sciences, pp. 4761-4770, IEEE, 2013.
- [28] Gomes Filho, A. F., Alencar, D., & Toledo, R. D., "Agile in 3D: Agility in the Animation Studio", *Brazilian Workshop on Agile Methods*, pp. 63-76, Springer, Cham, 2017.
- [29] Hirner, H., Lavicka, M., Schefer-Wenzl, S., & Miladinovic, I., "Agile Software Integration in Telecommunications—a Case Study", *27th Telecommunications Forum (TELFOR)*, pp. 1-4, IEEE, 2019.
- [30] Cruz, E. F. C. D., Fernandes Junior, F. E., & Sardinha, E. D., "An experience in the use of SCRUM and KANBAN for project development in a Waterfall environment", *XX Brazilian Symposium on Software Quality*, pp. 1-7, 2021.
- [31] Srivastava, A., Mehrotra, D., Kapur, P. K., & Aggarwal, A. G., "Analytical evaluation of agile success factors influencing quality in software industry", *International Journal of System Assurance Engineering and Management*, vol.11, no.2, pp.247-257, 2020.
- [32] da Cruz, A. F. et. al., "Blueprint model: An agile-oriented methodology for tackling global software development challenges", *Advances in Science Technology and Engineering Systems Journal*, vol.5, no.6, pp.353-362, 2020.
- [33] Gulliksen Stray, V., Moe, N. B., & Dingsøyr, T., "Challenges to teamwork: a multiple case study of two agile teams", *International conference on agile software development*, pp. 146-161, Springer, Berlin, Heidelberg, 2011.
- [34] Terlecka, K., "Combining Kanban and Scrum--Lessons from a team of sysadmins" *Agile Conference*, pp. 99-102, IEEE, 2012.
- [35] Saltz, J., & Crowston, K., "Comparing data science project management methodologies via a controlled experiment", *Hawaii International Conference on System Sciences*, 2017.
- [36] Granulo, A., & Tanović, A., "Comparison of SCRUM and KANBAN in the Learning Management System implementation process", *27th Telecommunications Forum (TELFOR)*, pp. 1-4, IEEE, 2019.
- [37] Shimoda, A., & Yabuki, T., "Cost and value analysis of software development method focused on individual function", *Eighth International Conference on Intelligent Computing and Information Systems (ICICIS)*, pp. 237-243, IEEE, 2017.
- [38] Nikitina, N., & Kajko-Mattsson, M., "Developer-driven big-bang process transition from Scrum to Kanban.", *International conference on software and systems process*, pp. 159-168, 2011.
- [39] Majchrzak, M., & Stilger, L., "Experience report: Introducing Kanban into automotive software project" *e-Informatica Software Engineering Journal*, vol. 11, no.1, 2017.
- [40] Birkeland, J. O., "From a timebox tangle to a more flexible flow", *International conference on agile software development*, pp. 325-334, Springer, Berlin, Heidelberg, 2010.
- [41] Rutherford, K., Shannon, P., Judson, C., & Kidd, N., "From chaos to kanban, via scrum", *International Conference on Agile Software Development*, pp. 344-352, Springer, Berlin, Heidelberg, 2010.
- [42] Gelmis, A., Ozkan, N., Ahmad, A. J., & Guler, M. G., "Impact of Turkish National Culture on Agile Software Development in Turkey", *International Conference on Lean and Agile Software Development*, pp. 78-95, Springer, Cham, 2022.
- [43] McKenzie, T., Morales-Trujillo, M., Lukosch, S., & Hoermann, S., "Is agile not agile enough? A study on how agile is applied and misapplied in the video game development industry", *IEEE/ACM Joint 15th International Conference on Software and System Processes (ICSSP) and 16th ACM/IEEE International Conference on Global Software Engineering (ICGSE)*, pp. 94-105, IEEE, 2021.
- [44] Seikola, M., & Loisa, H. M., "Kanban implementation in a telecom product maintenance", *37th EUROMICRO Conference on Software Engineering and Advanced Applications*, pp. 321-329, IEEE, 2011.
- [45] Raju, H. K., & Krishnegowda, Y. T., "Kanban Pull and Flow—A transparent workflow for improved quality and productivity in software developmet", *Fifth International Conference on Advances in Recent Technologies in Communication and Computing (ARTCom)*, pp. 44-51, IET, 2013.
- [46] Sjøberg, D. I., Johnsen, A., & Solberg, J., "Quantifying the effect of using kanban versus scrum: A case study", *IEEE software*, vol. 29, no. 5, pp.47-53, 2012.
- [47] Hašek, F., & Mohelská, H., "Selection of a Suitable Agile Methodology—Case Study," *Hradec Economic Days*, 2021.
- [48] Cocco, L., Mannaro, K., Concas, G., & Marchesi, M., "Simulating kanban and scrum vs. waterfall with system dynamics", *International conference on agile software development*, pp. 117-131, Springer, Berlin, Heidelberg, 2011.
- [49] Saltz, J., & Sutherland, A., "SKI: An agile framework for data science", *IEEE International Conference on Big Data (Big Data)*, pp. 3468-3476, IEEE, 2019.
- [50] Tudjarova, S., Chorbev, I., & Joksimoski, B., "Software Quality Metrics While Using Different Development Methodologies," *International Conference on ICT Innovations*, pp. 240-250, 2017.
- [51] Diebold, P., Theobald, S., Wahl, J., & Rausch, Y., "Stepwise transition to agile: From three agile practices to Kanban adaptation", *Journal of Software: Evolution and Process*, vol.31, no.5, e2167, 2019.
- [52] Senapathi, M., & Drury-Grogan, M. L., "Systems thinking approach to implementing kanban: A case study", *Journal of Software: Evolution and Process*, vol.33, no.4, e2322, 2021.
- [53] Iqbal, J., Omar, M., & Yasin, A., "The impact of agile methodologies and cost management success factors: An empirical study", *Baghdad Science Journal*, vol.16, no.2, pp.496-504, 2019.
- [54] Willeke, E. R., "The inkubook experience: A tale of five processes", *Agile Conference*, pp. 156-161, IEEE, 2009.
- [55] Ahmad, M. O., Kuvaja, P., Oivo, M., & Markkula, J., "Transition of software maintenance teams from Scrum to Kanban," *49th Hawaii International Conference on System Sciences (HICSS)*, pp. 5427-5436, IEEE, 2016.
- [56] Robinson, P. T., & Beecham, S., "TWINS-This Workflow Is Not Scrum: Agile process adaptation for open source software projects," *IEEE/ACM International Conference on Software and System Processes (ICSSP)*, pp. 24-33, IEEE, 2019.
- [57] Senapathi, M., & Srinivasan, A., "Understanding post-adoptive agile usage: An exploratory cross-case analysis", *Journal of Systems and Software*, vol.85, no.6, pp.1255-1268, 2012.
- [58] Lunesu, M. I., Münch, J., Marchesi, M., & Kuhrmann, M., "Using simulation for understanding and reproducing distributed software development processes in the cloud", *Information and Software Technology*, vol. 103, pp.226-238, 2018.
- [59] Anderson, D. J. *Kanban : successful evolutionary change in your technology business*. 2010.
- [60] Anderson, D. J., & Bozheva, T., *Kanban Maturity Model: Evolving Fit-For-Purpose Organizations: Lean Kanban*, 2018.

Software Requirements Classification using Deep-learning Approach with Various Hidden Layers

Sanidhya Vijayvargiya¹

Department of Computer Science & Information Systems
BITS Pilani Hyderabad Campus
f20202056@hyderabad.bits-pilani.ac.in

Lalita Bhanu Murthy³

Department of Computer Science & Information Systems
BITS Pilani Hyderabad Campus
bhanu@hyderabad.bits-pilani.ac.in

Lov Kumar²

Department of Computer Science & Information Systems
BITS Pilani Hyderabad Campus
lovkumar@hyderabad.bits-pilani.ac.in

Sanjay Misra⁴

Østfold University College, Halden, Norway
ssopam@gmail.com

Abstract—Software requirement classification is becoming increasingly crucial for the industry to keep up with the demand of growing project sizes. Based on client feedback or demand, software requirement classification is critical in segregating user needs into functional and quality requirements. However, because there are numerous machine learning (ML) and deep-learning (DL) models that require parameter tuning, the use of ML to facilitate decision-making across the software engineering pipeline is not well understood. Five distinct word embedding techniques were applied to the functional and quality software requirements in this study. The imbalanced classes in the dataset are balanced using Synthetic Minority Oversampling technique (SMOTE). Then, to reduce duplicate and unnecessary features, feature selection and dimensionality reduction techniques are used. Dimensionality reduction is accomplished with Principal Component Analysis (PCA), while feature selection is accomplished with the Rank-Sum Test (RST). For binary categorization into functional and non-functional needs, the generated vectors are provided as inputs to eight distinct Deep Learning classifiers. The findings of the research show that using a combination of word embedding and feature selection techniques in conjunction with various classifiers can accurately classify functional and quality software requirements.

Keywords—Functional Requirements, Non-Functional Requirements, Deep Learning, Data Imbalance Methods, Feature Selection, Classification Techniques, Word Embedding.

I. INTRODUCTION

SOFTWARE requirements classification deals with segregating the clients' requirements and demands found in the Software Requirements Specification (SRS) document into functional and non-functional requirements. It is a key step in the software development pipeline which can be automated using Machine Learning techniques. This allows the industry to save on labor expenses, as a domain expert is often required, while also optimizing the process and saving crucial time [1]. A key problem that needs to be addressed during requirements classification is that of the inconsistency in terminology used by the clients and the software engineers. This may lead to misclassification of the software requirements.

Functional requirements are the demands that the end-user defines as critical characteristics that the system should supply

and that can be observed immediately in the finished result. This is how the input to the system, the action to be taken, and the intended output are all defined or stated. The system's basic quality standards, often known as non-functional requirements, include factors like reliability, maintainability, security, and portability [2].

Another problem faced during software requirements classification is the imbalance between the number of instances of functional and non-functional requirements classes. Data imbalance means that the number of instances of minority class are much lower than those of the majority class. Because of the unbalanced distribution of data, classifiers are misled while learning the minority class, resulting in biased and erroneous findings [3]. A good software requirements categorization model will be one that has been trained on a similar distribution of functional and non-functional requirement classes. In this study, this problem is addressed using oversampling through Synthetic Minority Oversampling Technique (SMOTE).

In this paper, we look to solve the above problem, and create highly accurate software requirements classification models which can be reliably used in the industry. The following are the research questions (RQs) that will be used to attain the aforementioned goals.

- RQ1: Which feature extraction technique can best capture the unstructured, textual data present in the SRS document, and convert it to structured data in the form of numerical vectors?
- RQ2: Which feature selection techniques are the best at getting rid of redundant and irrelevant features which may affect the performance of the classification models?
- RQ3: For what structure of the deep learning classifiers do the software requirements classification models achieve the best results?
- RQ4: How does the application of class balancing technique through oversampling affect the performance of the models?

The criteria used to evaluate the performance of the various models are F-measure, accuracy, and Area under the ROC curve (AUC). The Friedman test was used to determine whether an ML technique resulted in a substantial difference in performance. The PROMISE dataset[4] was used in this study, which contains 625 labeled criteria from 15 different projects. The contributions of the study are as follows:

- An extensive comparison of various word embedding techniques for the purpose of feature extraction is provided to analyze which technique is best suited for the SRS document.
- A thorough investigation on the effect of various feature selection techniques on the performance of classification models in software engineering is presented.
- Deep learning classifiers are employed to increase the accuracy of software requirements classification from previous studies, with variations in number of layers, and type of layer being analyzed to find the best deep learning model out of the eight distinct DL classifiers used in this study.
- The study evaluates and analyses the performance of requirements classification models using relevant performance metrics. The study includes a thorough statistical analysis to back up the findings with statistical testing, unlike previous studies.
- The effect of class-balancing techniques on software datasets to build more accurate models is examined.

The remainder of the paper is structured as detailed here: Section II presents a literature review on software requirement classification and various word embedding approaches that are used in this study. Section III describes the experimental dataset collection as well as the various machine learning algorithms used. The research methodology is described in Section IV using an architecture framework. In Section V, the results of the experiments, along with their analysis, are presented. Section VI shows a comparison of models created using various word-embedding approaches, sets of features, and machine learning models. Finally, Section VII summarizes the information provided and offers directions for further research.

II. RELATED WORK

A. Software Requirements Word Embeddings

Navarro-Almanza et al. [5] explore Word2Vec using Skip-Gram to get structured representation for the textual software requirements dataset. The purpose of the Skip-gram model is to anticipate context words. The projection from Skip-Gram is a continuous vector space rendition of the word with a low dimensionality. The models developed achieved a maximum of 0.8 precision, 0.785 recall, and 0.77 F-measure.

To improve how well the word embeddings capture the content of the text, Marcacini et al. [6] analyze the impact of using contextual word embeddings for software requirements. They use the RE-BERT model to obtain the structured data to feed into their hierarchical clustering classifier. BERT is

built on the Transformers architecture and uses a deep neural network. To address the sequence of occurrence of the tokens, a positional embedding is used. Static word embeddings, such as word2vec and FastText, on the other hand, have the issue of having the same embedding regardless of context, which makes structured modeling of software requirements difficult.

B. Classifying Software Requirements

Ott [7] employ two classifiers, Multinomial Naive Bayes, and Support Vector machine for software requirements classification. The classification techniques are applied on two datasets, out of which one is confidential and the other is public. The maximum recall achieved by the Naive Bayes classifier is 0.94, whereas the Support Vector machine achieves a precision of 0.86 in the best model. Baker et al. [8] work on classification of non-functional requirements into their sub-categories. The authors compare the performance of the CNN model with that of an ANN model and results indicate that the CNN model outperforms the ANN on most performance metrics. The ANN consists of one hidden layer of 20 neurons. The ANN model has a precision of 82% to 90%, a recall of 78% to 85%, and the greatest F-score of 84%. With the maximum F-score of 92%, the CNN model obtains precision between 82% and 94%, and recall between 76% and 97%. Rahimi et al. [9] focus on further classifying the functional requirements into six different categories: policy, action constraint, solution, definition, attribute constraint, and enablement. The authors use the ensemble approach which combined five distinct classifiers: support vector classification, support vector machine, decision tree, logistic regression, and Naive Bayes. For each class, accuracy per class as a weight is used to find the most optimal classifier. The best results are obtained using LR, SVC, SVM as the classifiers, which perform better than using all classifiers. An accuracy of 99.45% is achieved in classifying 600 functional requirements.

III. STUDY DESIGN

This section presents the details regarding various design setting used for this research.

A. Experimental Dataset

Cleland-Huang and his team [4] used the same datasets to validate the proposed software requirement solution. Cleland-Huang and his team extracted this data with the help of MS students from DePaul University and rendered it for public research via the PROMISE repository. The functional and non-functional attributes are shown below in Figure 2. The first observation to be made about Figure 2 is that the PROMISE repository is uneven in number of functional and non-functional requirements, with quality requirements accounting for 382 of the 625 total.

B. Training of Models from Imbalanced Data Set:

Several ML algorithms have the issue of neglecting the minority class in unbalanced datasets, despite the reality that performance on those is often what matters. In order to use future ML techniques, it was essential to implement the SMOTE technique on the imbalanced classes in our dataset. SMOTE [10] (Synthetic Minority Oversampling Approach) is

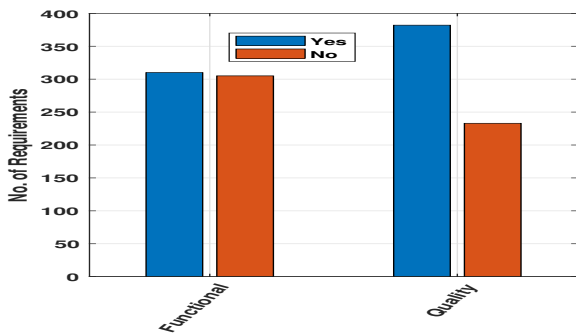


Fig. 2: Data-Sets

a data augmentation technique that duplicates existing minority class instances or generates new minority class instances. It solves the imbalance problem without adding any new data to the dataset, allowing us to employ machine learning techniques afterwards.

C. Word Embedding:

The dataset’s textual data must be expressed as vectors in respect to one another. The dataset was subjected to five different word embedding techniques: Term Frequency and Inverse Document Frequency (TF-IDF), Continuous Bag of Words (CBOW)¹, Skip-Gram (SKG), Global Vectors for Word Representation (GLOVE)², and Google news Word to Vector (GW2V). The aim of these techniques, such as GLOVE, CBOW, etc., is to capture the semantic information in the text, which is not possible with other word-embedding techniques such as TF-IDF. The textual data was represented as a vector in an n-dimensional space using these techniques. Before applying the word embedding techniques, all characters in the requirements are converted to lowercase letters. We deleted any stopwords, bad symbols, and spaces. These will now be

¹<https://towardsdatascience.com/word2vec-skip-gram-model-part-1-intuition-78644e111111>

²<https://nlp.stanford.edu/projects/glove/>

utilized to create models that will classify the requirements into functional and non-functional [11].

D. Feature Selection Techniques

We identify critical feature vectors that impact the performance of the models after doing word embedding on the data. To eliminate redundant and unnecessary features that may have a detrimental impact on the models’ performance, the Rank Sum Test (RST) and Principal Component Analysis (PCA) are used. To see the difference, the performance of models with these features is compared to the performance of models without these features. This phase aids in the reduction of over-training and training time [12].

E. Classification Technique:

The dataset is divided into training and testing subsets and categorized using eight deep learning models using K-Fold Cross-validation with a k value of 5. The structure of the various models is presented below. All models have an input layer with number neurons equal to the number of features of input data. For each subsequent hidden layer, the number of neurons is halved. All layers involved in the Deep Learning models are either Dense layers or Dropout layers. In a Dense layer, each neuron in the layer receives input from all neurons of the previous layer. On the other hand, a Dropout layer randomly selects and omits a certain number of neurons in the layer while training the Deep Learning model. In this work, a dropout value of 0.2 is used. Dropout layers are used to solve the problem of overfitting models. Finally, the output layer consists of only one neuron which contains a binary value corresponding to the binary classification to either functional or non-functional requirements. The activation function for each layer is the rectified linear activation function or ReLU, except the output layer which uses a sigmoid activation function. Binary cross entropy is the loss function that is utilized to train the models with Adam as the optimizer. Figure 3 shows the architecture of the considered deep learning model1, model2, model3, and model4 (DL1, DL2, DL3, and DL4). Similarly, we

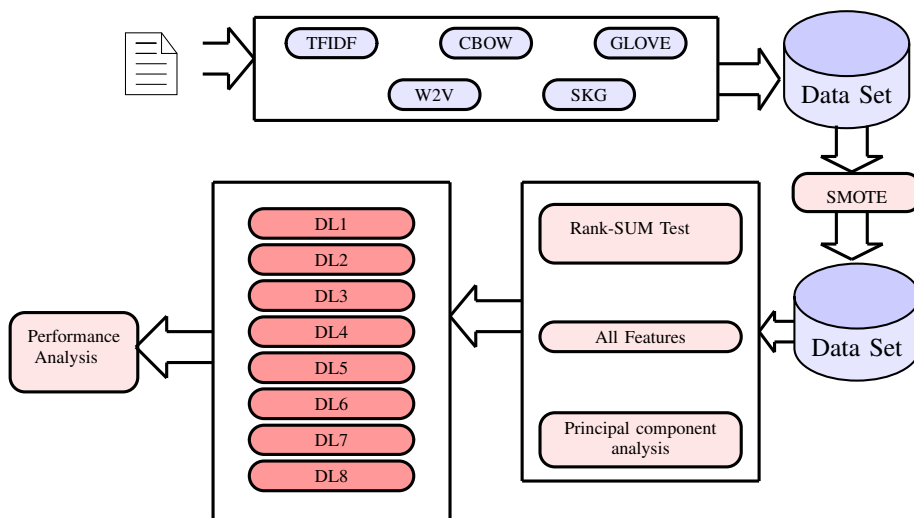


Fig. 1: Framework of proposed work

are increasing the number of hidden layers for four more deep learning models. The above considered models are validated using 5-fold cross-validation with batch size=30, epochs=100, and Dropout=0.2.

IV. RESEARCH METHODOLOGY

Figure 1 provides a full outline of our proposed effort. We started by extracting the software requirements dataset from the PROMISE repository. This data contains labels of the software requirement category it falls under, i.e. functional or non-functional requirements. The data was then subjected to pre-processing techniques like converting all characters to lowercase, removing non-alphanumeric and non-symbol characters, eliminating frequently used words like 'the', and 'a', and other words with lengths less than or equal to two as they do not make a significant impact in the classification. Then, all of the phrases were tokenized to words.

To extract features from this unstructured, pre-processed data, five different word embedding techniques were applied to best capture the information. To account for the imbalance in classes due to 382 instances of quality requirements out of 625, a data oversampling technique in the form of SMOTE was used. Models were trained on both the balanced and imbalanced datasets to compare the effectiveness of the class-balancing. The features acquired after the word embeddings and class balancing needed to be refined to remove redundant and irrelevant features. Feature selection technique in the form of Rank-Sum test, and dimensionality reduction technique in the form of Principal Component Analysis were employed. These two sets of features as well as the set of original features were fed to the Deep Learning classifiers. This was done to help understand the importance of feature selection.

The eight different DL classifiers were trained using 5-fold cross validation. The classifiers have varying layer sizes, and layer types, but certain attributes like the optimizer, loss function, etc. remain constant across the different classifiers. These classifiers are named DL1, DL2, DL3, and so on till DL8. Finally, the performance of the various models developed was measured using accuracy, F-measure, and AUC. This performance was statistically analyzed using box-plots for visual representation, with statistics like mean, maximum, minimum, Q1, and Q3 for each performance metric. Further, any conclusions derived were statistically supported using the Friedman test.

V. EMPIRICAL RESULTS AND ANALYSIS

To categorize software needs into functional or non-functional, we used five distinct word embedding approaches, a class balance strategy, two feature selection strategies, and eight different classification techniques. As a result, a total of 480 [5 word-embedding techniques \times 2 requirements datasets (1 functional requirements dataset + 1 non-functional requirements dataset) \times (1 imbalanced dataset + 1 balanced dataset) \times 3 sets of features \times 8 DL classifiers] models were generated. As shown in Tables I and II, the predictive performance of these trained models is assessed using the F-measure, accuracy and Area Under Curve (AUC) performance metrics.

- The high value of AUC confirms that the developed models have the ability to accurately classify the software requirements into functional and non-functional as almost all the performance values seen on the right side of Table I are greater than 0.75 AUC.
- The models developed using the Deep Learning structure of DL3 have better performance as compared to other classifiers.
- The models trained using neural network with ADAM (NNADAM) training algorithm have better predictive ability as compared LBMF, and SGD training algorithms.
- Simply by observing the values in Table I, we can see the difference SMOTE provides in improving the performance.

VI. COMPARATIVE ANALYSIS

The various models created with using word embedding techniques, class balancing approaches, feature selection strategies, and different classifiers are compared in this section. The comparison is based on statistics such as the area under the ROC curve (AUC), F-measure, and accuracy, with box plots serving as a visualization of the comparative performance. The Friedman test was used in this research to verify the findings. The Friedman test is used to accept or reject the following hypothesis.

- **Null Hypothesis**- There is no substantial difference in the predictive performance of software classification models constructed using different machine learning approaches.
- **Alternate Hypothesis**- The prediction power of software classification models constructed using various ML approaches varies significantly.

With degrees of freedom of 4 for word embedding, 1 for class balancing, 2 for feature selection, and 7 for distinct DL classifier comparisons, the Friedman test was performed with a significance threshold of $\alpha = 0.05$.

A. RQ1: Which feature extraction technique can best capture the unstructured, textual data present in the SRS document, and convert it to structured data in the form of numerical vectors?

In this work, five distinct word embedding approaches were utilized to compute the numerical vector of the functional and quality requirements: TF-IDF, Skip-Gram (SKG), Global Vectors for Word Representation (GloVe), W2V and Bag of Words (CBOW). To assess the prediction abilities of models generated using various word embedding techniques, the AUC, accuracy, and F-measure statistics were used.

1) Box-plot: Word-Embedding: Figure 4 illustrates the result of several word embedding algorithms. Models developed with the word embedding generated by TF-IDF are more reliable than other models, as shown in Figure 4. CBoW models exhibit poor prediction performance when compared to other methodologies. The mean AUC value of TF-IDF models is 0.91, with a maximum AUC of 0.98 and a Q3 accuracy of 0.96, implying that 25% of TF-IDF models have an AUC value

Model: "sequential_13"			Model: "sequential_14"		
Layer (type)	Output Shape	Param #	Layer (type)	Output Shape	Param #
dense_36 (Dense)	(None, 1605)	2577630	dense_38 (Dense)	(None, 1605)	2577630
dense_37 (Dense)	(None, 1)	1606	dropout_8 (Dropout)	(None, 1605)	0
Total params: 2,579,236			Total params: 2,579,236		
Trainable params: 2,579,236			Trainable params: 2,579,236		
Non-trainable params: 0			Non-trainable params: 0		

(3.1) DL1

Model: "sequential_16"			Model: "sequential_12"		
Layer (type)	Output Shape	Param #	Layer (type)	Output Shape	Param #
dense_43 (Dense)	(None, 1605)	2577630	dense_31 (Dense)	(None, 1605)	2577630
dropout_9 (Dropout)	(None, 1605)	0	dense_32 (Dense)	(None, 802)	1288012
dense_44 (Dense)	(None, 802)	1288012	dense_33 (Dense)	(None, 401)	322003
dense_45 (Dense)	(None, 1)	803	dense_34 (Dense)	(None, 401)	161202
Total params: 3,866,445			Total params: 4,349,249		
Trainable params: 3,866,445			Trainable params: 4,349,249		
Non-trainable params: 0			Non-trainable params: 0		

(3.3) DL4

(3.2) DL2

(3.4) DL8

Fig. 3: Deep Learning Architecture

of more than 0.96. The accuracy data and F-measure of these models confirm these findings, revealing that classification techniques based on TF-IDF outperform classification models based on alternative word-embedding techniques.

2) *Friedman Test: Word-Embedding:* In this work, the Friedman Test is also utilized to examine the predictive power of the models created using different word embedding methods. The goal of the test is to see if the null hypothesis is correct. The null hypothesis asserts that "the various word embedding techniques have no discernible impact on the performance of the classification models." Table IV shows the mean ranks for the various word embedding techniques. The lower the mean rank, the better the models' performance. TF-IDF has the lowest mean rank of 1.88, while CBoW has the highest mean rank of 4.88. With a mean rank of 1.96, W2V provides comparable performance to TF-IDF. It's probable that the high performance of the TF-IDF is due to the fact that requirements papers contain numerous comparable phrases and terms. Because of its method of giving weights to each term, TF-IDF can effectively minimize frequent terms used in requirements from creating an effect on classification better than other word-embedding algorithms.

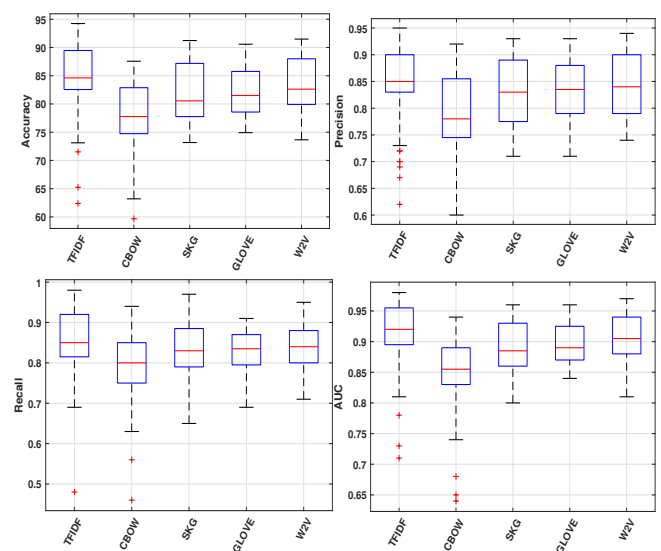


Fig. 4: Performance Box-Plot Diagram: Performance of Different Word Embedding

TABLE I: Precision and Recall

		Precision								Recall							
		OD(AF)															
		DL1	DL2	DL3	DL4	DL5	DL6	DL7	DL8	DL1	DL2	DL3	DL4	DL5	DL6	DL7	DL8
TFIDF	FUN	0.85	0.84	0.85	0.85	0.85	0.83	0.84	0.87	0.8	0.84	0.85	0.86	0.85	0.87	0.85	0.85
TFIDF	QUA	0.8	0.93	0.93	0.93	0.93	0.93	0.94	0.92	0.98	0.93	0.92	0.92	0.92	0.91	0.9	0.94
CBOW	FUN	0.74	0.76	0.78	0.76	0.75	0.72	0.76	0.8	0.73	0.81	0.76	0.8	0.76	0.8	0.79	0.73
CBOW	QUA	0.74	0.87	0.88	0.86	0.87	0.85	0.84	0.87	0.94	0.83	0.85	0.88	0.86	0.86	0.77	0.84
SKG	FUN	0.82	0.76	0.79	0.79	0.76	0.79	0.82	0.76	0.69	0.79	0.79	0.75	0.81	0.75	0.77	0.78
SKG	QUA	0.85	0.89	0.89	0.88	0.88	0.89	0.84	0.88	0.91	0.9	0.88	0.86	0.93	0.87	0.94	0.84
GLOVE	FUN	0.8	0.79	0.79	0.78	0.8	0.76	0.8	0.78	0.74	0.82	0.82	0.81	0.8	0.82	0.82	0.79
GLOVE	QUA	0.85	0.88	0.87	0.84	0.88	0.88	0.84	0.85	0.87	0.84	0.87	0.9	0.86	0.84	0.89	0.87
W2V	FUN	0.8	0.78	0.81	0.83	0.78	0.81	0.8	0.82	0.83	0.79	0.79	0.75	0.81	0.77	0.75	0.79
W2V	QUA	0.9	0.91	0.87	0.91	0.91	0.89	0.89	0.89	0.88	0.87	0.89	0.87	0.87	0.9	0.89	0.85
		OD(RST)															
TFIDF	FUN	0.88	0.87	0.84	0.85	0.86	0.84	0.86	0.85	0.75	0.82	0.82	0.81	0.8	0.81	0.8	0.8
TFIDF	QUA	0.81	0.91	0.9	0.9	0.9	0.9	0.9	0.88	0.98	0.92	0.91	0.93	0.93	0.92	0.92	0.93
CBOW	FUN	0.73	0.79	0.77	0.74	0.77	0.81	0.75	0.8	0.82	0.79	0.78	0.84	0.79	0.7	0.79	0.75
CBOW	QUA	0.77	0.87	0.82	0.86	0.84	0.84	0.87	0.82	0.92	0.86	0.92	0.87	0.88	0.88	0.85	0.92
SKG	FUN	0.78	0.76	0.75	0.79	0.78	0.75	0.82	0.77	0.77	0.83	0.83	0.82	0.78	0.8	0.77	0.79
SKG	QUA	0.84	0.89	0.86	0.88	0.89	0.88	0.9	0.89	0.9	0.9	0.91	0.91	0.91	0.89	0.9	0.88
GLOVE	FUN	0.78	0.79	0.81	0.8	0.83	0.79	0.79	0.79	0.79	0.83	0.8	0.8	0.76	0.81	0.79	0.82
GLOVE	QUA	0.84	0.85	0.9	0.9	0.88	0.87	0.88	0.87	0.87	0.89	0.85	0.87	0.89	0.9	0.88	0.88
W2V	FUN	0.81	0.81	0.75	0.85	0.83	0.81	0.79	0.79	0.83	0.82	0.84	0.75	0.77	0.8	0.82	0.82
W2V	QUA	0.87	0.88	0.9	0.9	0.9	0.88	0.88	0.9	0.89	0.93	0.9	0.91	0.88	0.88	0.91	0.9
		OD(PCA)															
TFIDF	FUN	0.69	0.77	0.75	0.78	0.7	0.76	0.76	0.7	0.79	0.75	0.77	0.7	0.82	0.78	0.72	0.79
TFIDF	QUA	0.74	0.88	0.83	0.9	0.84	0.85	0.85	0.86	0.94	0.83	0.9	0.85	0.88	0.85	0.88	0.85
CBOW	FUN	0.63	0.66	0.69	0.67	0.66	0.71	0.68	0.68	0.7	0.69	0.78	0.72	0.67	0.73	0.67	0.78
CBOW	QUA	0.64	0.76	0.82	0.78	0.85	0.82	0.76	0.81	0.91	0.82	0.85	0.84	0.67	0.74	0.85	0.83
SKG	FUN	0.76	0.79	0.75	0.75	0.81	0.74	0.8	0.74	0.68	0.79	0.78	0.85	0.79	0.82	0.75	0.82
SKG	QUA	0.71	0.89	0.9	0.9	0.85	0.89	0.89	0.89	0.97	0.87	0.88	0.86	0.9	0.85	0.85	0.86
GLOVE	FUN	0.8	0.77	0.77	0.8	0.75	0.8	0.77	0.8	0.69	0.78	0.77	0.78	0.8	0.79	0.79	0.75
GLOVE	QUA	0.75	0.85	0.88	0.87	0.86	0.87	0.84	0.87	0.91	0.87	0.86	0.85	0.85	0.88	0.91	0.86
W2V	FUN	0.78	0.78	0.76	0.77	0.77	0.76	0.81	0.77	0.71	0.8	0.83	0.81	0.79	0.78	0.74	0.77
W2V	QUA	0.79	0.88	0.88	0.88	0.88	0.87	0.87	0.88	0.95	0.86	0.86	0.86	0.87	0.86	0.87	0.83
		SMOTE(AF)															
TFIDF	FUN	0.84	0.84	0.84	0.84	0.85	0.83	0.84	0.81	0.8	0.84	0.85	0.85	0.85	0.84	0.84	0.84
TFIDF	QUA	0.88	0.94	0.94	0.95	0.95	0.95	0.95	0.94	0.92	0.92	0.92	0.93	0.93	0.93	0.92	0.93
CBOW	FUN	0.78	0.75	0.75	0.8	0.7	0.76	0.66	0.77	0.46	0.8	0.76	0.73	0.86	0.76	0.78	0.65
CBOW	QUA	0.86	0.89	0.89	0.77	0.86	0.89	0.85	0.89	0.69	0.85	0.85	0.86	0.85	0.84	0.8	0.81
SKG	FUN	0.75	0.8	0.77	0.71	0.78	0.73	0.77	0.73	0.84	0.71	0.74	0.82	0.8	0.81	0.83	0.81
SKG	QUA	0.87	0.9	0.92	0.85	0.93	0.89	0.91	0.87	0.87	0.85	0.89	0.9	0.82	0.87	0.91	0.93
GLOVE	FUN	0.81	0.81	0.78	0.79	0.77	0.81	0.81	0.79	0.78	0.82	0.85	0.77	0.8	0.81	0.73	0.82
GLOVE	QUA	0.88	0.92	0.92	0.9	0.92	0.92	0.93	0.87	0.84	0.85	0.86	0.88	0.86	0.86	0.81	0.89
W2V	FUN	0.79	0.83	0.83	0.82	0.75	0.81	0.79	0.79	0.73	0.82	0.78	0.81	0.89	0.81	0.8	0.81
W2V	QUA	0.9	0.92	0.92	0.88	0.91	0.91	0.91	0.8	0.87	0.88	0.87	0.89	0.91	0.93	0.87	0.92
		SMOTE(RST)															
TFIDF	FUN	0.85	0.85	0.85	0.82	0.83	0.83	0.85	0.85	0.73	0.83	0.82	0.84	0.83	0.83	0.84	0.82
TFIDF	QUA	0.85	0.92	0.91	0.92	0.92	0.9	0.93	0.92	0.92	0.9	0.91	0.9	0.91	0.91	0.91	0.93
CBOW	FUN	0.77	0.79	0.8	0.77	0.76	0.78	0.79	0.78	0.77	0.79	0.75	0.8	0.82	0.75	0.8	0.78
CBOW	QUA	0.79	0.86	0.84	0.86	0.88	0.92	0.89	0.86	0.88	0.87	0.87	0.9	0.86	0.83	0.83	0.86
SKG	FUN	0.75	0.79	0.8	0.83	0.78	0.79	0.77	0.75	0.81	0.81	0.83	0.75	0.8	0.81	0.82	0.84
SKG	QUA	0.86	0.91	0.92	0.93	0.88	0.87	0.9	0.93	0.86	0.85	0.83	0.79	0.9	0.89	0.85	0.83
GLOVE	FUN	0.78	0.81	0.83	0.79	0.84	0.79	0.77	0.71	0.79	0.83	0.79	0.78	0.78	0.81	0.82	0.85
GLOVE	QUA	0.86	0.9	0.91	0.91	0.9	0.91	0.87	0.91	0.85	0.88	0.86	0.85	0.87	0.9	0.89	0.86
W2V	FUN	0.79	0.8	0.79	0.79	0.85	0.82	0.83	0.78	0.83	0.84	0.83	0.81	0.78	0.84	0.77	0.8
W2V	QUA	0.88	0.92	0.93	0.93	0.93	0.92	0.9	0.88	0.88	0.85	0.87	0.88	0.87	0.88	0.88	0.91
		SMOTE(PCA)															
TFIDF	FUN	0.67	0.76	0.72	0.74	0.72	0.74	0.62	0.73	0.48	0.69	0.79	0.77	0.79	0.77	0.77	0.8
TFIDF	QUA	0.86	0.88	0.91	0.87	0.88	0.81	0.9	0.92	0.87	0.92	0.86	0.84	0.93	0.92	0.83	0.86
CBOW	FUN	0.6	0.7	0.71	0.74	0.73	0.7	0.68	0.77	0.56	0.69	0.85	0.78	0.71	0.79	0.69	0.75
CBOW	QUA	0.73	0.85	0.87	0.87	0.88	0.81	0.77	0.88	0.63	0.78	0.81	0.72	0.77	0.84	0.81	0.77
SKG	FUN	0.78	0.76	0.78	0.78	0.76	0.83	0.8	0.79	0.65	0.82	0.77	0.81	0.76	0.69	0.73	0.74
SKG	QUA	0.85	0.88	0.92	0.9	0.92	0.9	0.9	0.88	0.88	0.9	0.87	0.87	0.9	0.9	0.92	0.9
GLOVE	FUN	0.74	0.77	0.77	0.76	0.74	0.78	0.78	0.76	0.77	0.78	0.78	0.8	0.83	0.77	0.77	0.8
GLOVE	QUA	0.87	0.9	0.89	0.89	0.89	0.9	0.91	0.91	0.82	0.85	0.88	0.89	0.89	0.87	0.88	0.85
W2V	FUN	0.74	0.79	0.78	0.78	0.8	0.8	0.78	0.74	0.73	0.83	0.74	0.84	0.78	0.78	0.77	0.84
W2V	QUA	0.87	0.92	0.91	0.91	0.92	0.94	0.91	0.89	0.87	0.87	0.87	0.87	0.88	0.83	0.87	0.9

B. RQ2: Which feature selection techniques are the best at getting rid of redundant and irrelevant features which may affect the performance of the classification models?

In the proposed study, we use Rank Sum test and PCA as feature selection procedures, and we use all of the original

features for developing predictive models for requirements categorization in a third set of models. These feature selection procedures were applied to both the functional and quality requirements datasets.

TABLE II: Accuracy and AUC

		Accuracy								AUC							
		DL1	DL2	DL3	DL4	DL5	DL6	DL7	DL8	DL1	DL2	DL3	DL4	DL5	DL6	DL7	DL8
		OD(AF)															
TFIDF	FUN	83.2	84.32	85.12	85.60	85.12	84.80	84.80	85.92	0.9	0.91	0.92	0.92	0.92	0.91	0.91	0.91
TFIDF	QUA	83.52	91.36	91.04	91.20	90.88	90.40	90.40	91.20	0.93	0.96	0.96	0.96	0.96	0.96	0.96	0.95
CBOW	FUN	73.6	77.44	77.12	77.44	76.00	74.88	76.80	77.44	0.8	0.85	0.84	0.85	0.84	0.83	0.84	0.84
CBOW	QUA	76	81.92	83.68	83.68	83.36	82.40	76.96	82.88	0.83	0.89	0.9	0.88	0.89	0.89	0.84	0.89
SKG	FUN	77.12	77.60	79.04	77.76	77.76	77.76	80.16	77.12	0.86	0.86	0.86	0.86	0.85	0.87	0.87	0.85
SKG	QUA	84.16	87.52	85.92	84.64	87.68	84.96	85.44	83.68	0.91	0.93	0.93	0.92	0.92	0.91	0.92	0.91
GLOVE	FUN	78.08	80.48	80.32	79.20	80.16	78.24	80.96	78.88	0.86	0.87	0.87	0.86	0.87	0.87	0.85	0.87
GLOVE	QUA	82.88	83.20	84.32	83.04	84.32	82.88	82.88	82.4	0.9	0.91	0.92	0.9	0.91	0.9	0.91	0.89
W2V	FUN	81.44	78.88	80.64	80.00	79.36	79.84	78.08	80.64	0.89	0.88	0.88	0.88	0.87	0.86	0.87	0.87
W2V	QUA	86.72	87.04	85.60	87.04	86.72	87.52	86.56	84.8	0.92	0.93	0.93	0.94	0.93	0.93	0.93	0.92
		OD(RST)															
TFIDF	FUN	82.56	84.80	83.20	83.36	83.52	82.88	83.84	83.36	0.89	0.92	0.92	0.92	0.91	0.91	0.91	0.91
TFIDF	QUA	84.64	89.28	88.80	89.12	89.28	88.64	89.28	88.16	0.93	0.95	0.95	0.95	0.95	0.94	0.95	0.93
CBOW	FUN	76	79.04	77.44	77.44	77.92	76.96	76.48	78.24	0.84	0.87	0.86	0.86	0.86	0.86	0.85	0.86
CBOW	QUA	78.08	83.68	82.88	82.88	82.72	82.24	83.20	82.88	0.86	0.9	0.9	0.89	0.91	0.85	0.9	0.91
SKG	FUN	77.60	78.72	77.92	80.32	78.08	76.8	80.32	77.60	0.85	0.88	0.86	0.86	0.86	0.85	0.86	0.85
SKG	QUA	83.04	86.88	85.60	87.20	87.36	85.92	87.20	86.08	0.9	0.92	0.92	0.92	0.92	0.91	0.92	0.91
GLOVE	FUN	78.24	80.32	80.80	80.32	80.48	79.84	79.20	80.48	0.86	0.88	0.88	0.88	0.88	0.87	0.87	0.88
GLOVE	QUA	81.92	83.84	84.80	86.24	85.76	85.76	85.28	84.80	0.89	0.91	0.91	0.92	0.92	0.92	0.92	0.9
W2V	FUN	81.44	81.76	78.24	81.12	80.96	80.64	80.16	80.16	0.89	0.9	0.89	0.9	0.9	0.89	0.88	0.89
W2V	QUA	85.12	88.16	87.84	88.32	86.72	85.76	86.88	87.36	0.93	0.94	0.94	0.94	0.93	0.93	0.93	0.93
		OD(PCA)															
TFIDF	FUN	71.52	76.80	75.84	75.04	73.60	76.64	74.56	73.12	0.78	0.85	0.84	0.83	0.82	0.84	0.81	0.82
TFIDF	QUA	76.16	82.56	82.56	85.12	82.24	81.60	83.36	82.72	0.82	0.91	0.9	0.92	0.89	0.89	0.85	0.89
CBOW	FUN	64.48	67.04	71.84	68.80	66.56	71.52	67.84	70.56	0.68	0.75	0.79	0.77	0.74	0.78	0.74	0.77
CBOW	QUA	63.2	73.60	79.36	76.16	72.64	74.24	74.24	78.08	0.65	0.81	0.85	0.83	0.81	0.82	0.8	0.85
SKG	FUN	73.6	79.04	76.64	78.40	80.00	76.96	78.56	76.48	0.8	0.86	0.85	0.87	0.86	0.86	0.87	0.86
SKG	QUA	74.56	85.44	86.40	86.08	84.64	84.16	84.32	84.48	0.89	0.92	0.93	0.92	0.91	0.91	0.91	0.92
GLOVE	FUN	76.16	77.12	76.96	79.36	76.80	79.68	77.76	78.08	0.84	0.85	0.84	0.85	0.85	0.86	0.84	0.85
GLOVE	QUA	76.48	82.40	84.32	83.04	82.24	84.80	84.00	83.52	0.86	0.9	0.91	0.9	0.9	0.91	0.9	0.89
W2V	FUN	75.84	78.72	78.72	78.40	77.92	77.12	78.40	77.12	0.84	0.85	0.86	0.86	0.86	0.86	0.84	0.84
W2V	QUA	81.6	84.80	84.16	84.64	84.96	83.68	83.84	82.72	0.9	0.92	0.92	0.92	0.92	0.92	0.9	0.91
		SMOTE(AF)															
TFIDF	FUN	82.22	84.13	84.29	84.60	85.08	83.65	84.29	82.22	0.89	0.91	0.91	0.91	0.91	0.91	0.9	0.89
TFIDF	QUA	90.05	93.32	93.19	93.98	94.11	94.24	93.59	93.32	0.96	0.98	0.98	0.98	0.98	0.98	0.98	0.98
CBOW	FUN	66.51	76.35	75.40	77.62	74.60	76.03	69.05	73.17	0.79	0.84	0.83	0.83	0.84	0.84	0.77	0.83
CBOW	QUA	78.93	87.17	87.30	79.97	85.60	86.91	83.12	85.47	0.9	0.92	0.92	0.88	0.92	0.92	0.89	0.92
SKG	FUN	77.62	76.83	76.03	74.44	78.73	75.71	79.21	75.71	0.85	0.85	0.83	0.84	0.86	0.84	0.85	0.83
SKG	QUA	87.17	87.83	90.31	87.04	88.35	88.22	90.71	89.66	0.93	0.95	0.96	0.93	0.96	0.94	0.95	0.95
GLOVE	FUN	80.16	81.43	80.63	78.25	77.94	80.95	78.25	80.16	0.87	0.89	0.88	0.87	0.87	0.89	0.87	0.87
GLOVE	QUA	86.39	88.61	89.14	88.87	89.14	88.87	87.83	88.09	0.93	0.95	0.96	0.95	0.95	0.94	0.94	0.94
W2V	FUN	76.67	82.54	80.95	81.43	79.84	80.79	79.21	79.37	0.85	0.9	0.89	0.9	0.9	0.88	0.88	0.87
W2V	QUA	88.48	89.92	89.79	88.48	90.97	91.49	89.01	84.16	0.95	0.96	0.96	0.95	0.96	0.96	0.96	0.94
		SMOTE(RST)															
TFIDF	FUN	80.32	83.81	84.13	82.86	83.49	83.17	84.60	83.81	0.9	0.92	0.92	0.92	0.92	0.91	0.92	0.91
TFIDF	QUA	87.43	90.97	91.10	91.23	91.62	90.71	92.02	92.41	0.94	0.97	0.97	0.97	0.97	0.97	0.97	0.97
CBOW	FUN	76.51	79.05	77.94	77.94	78.10	76.67	79.21	78.10	0.84	0.87	0.86	0.85	0.87	0.86	0.87	0.86
CBOW	QUA	82.2	86.39	85.47	87.57	87.30	87.57	86.13	86.26	0.89	0.93	0.93	0.93	0.93	0.94	0.92	0.92
SKG	FUN	76.83	79.68	80.79	79.52	78.41	79.52	78.89	78.41	0.85	0.88	0.88	0.87	0.87	0.88	0.87	0.86
SKG	QUA	86.13	88.35	88.09	86.65	89.27	88.22	87.43	88.35	0.92	0.95	0.96	0.95	0.95	0.95	0.95	0.95
GLOVE	FUN	78.57	81.59	81.43	78.57	81.27	79.52	78.57	75.56	0.86	0.88	0.89	0.88	0.88	0.87	0.87	0.84
GLOVE	QUA	85.47	89.01	88.74	88.22	88.61	90.58	87.96	88.87	0.93	0.94	0.95	0.94	0.95	0.95	0.94	0.93
W2V	FUN	80.16	81.75	80.48	80.00	82.06	82.70	80.63	78.57	0.89	0.9	0.91	0.9	0.91	0.9	0.9	0.88
W2V	QUA	88.22	88.48	90.05	90.97	90.18	90.31	89.14	89.27	0.94	0.96	0.97	0.97	0.97	0.97	0.96	0.95
		SMOTE(PCA)															
TFIDF	FUN	62.38	73.97	74.44	75.08	73.81	74.92	65.24	74.76	0.71	0.83	0.82	0.84	0.82	0.83	0.73	0.82
TFIDF	QUA	86.78	89.66	89.14	85.99	90.05	85.08	86.65	88.87	0.94	0.96	0.96	0.94	0.96	0.93	0.93	0.95
CBOW	FUN	59.68	69.52	75.56	75.40	72.22	72.54	68.25	76.19	0.64	0.77	0.84	0.83	0.8	0.81	0.76	0.81
CBOW	QUA	70.16	82.07	84.29	80.50	82.98	82.07	78.80	83.51	0.76	0.89	0.91	0.89	0.88	0.89	0.86	0.9
SKG	FUN	73.17	77.78	77.94	79.05	76.19	77.30	77.30	77.30	0.81	0.85	0.85	0.86	0.85	0.85	0.85	0.84
SKG	QUA	86.13	88.87	89.79	88.87	91.10	90.31	91.23	89.14	0.93	0.95	0.95	0.95	0.96	0.95	0.95	0.95
GLOVE	FUN	74.92	77.30	77.30	77.78	76.98	77.78	77.46	77.30	0.84	0.86	0.86	0.86	0.85	0.86	0.85	0.84
GLOVE	QUA	84.95	87.57	88.48	89.14	89.27	88.74	89.27	87.96	0.93	0.94	0.94	0.94	0.94	0.94	0.94	0.93
W2V	FUN	73.65	80.32	76.83	80.16	78.89	79.21	77.62	77.30	0.81	0.88	0.86	0.88	0.87	0.87	0.86	0.86
W2V	QUA	86.91	89.53	89.40	88.87	90.18	89.01	89.01	89.14	0.94	0.96	0.95	0.96	0.96	0.96	0.95	0.95

1) *Box-plot: Feature Selection:* RST seems to select a better subset of features than any other technique, per Figure 5. RST has an average AUC of 0.91, with a lowest of 0.84 and a peak of 0.97. The features created using PCA performed the worst of the three sets of features, with a mean AUC of 0.87.

2) *Friedman Test: Feature Selection:* We also utilized the Friedman test to evaluate the various feature selection procedures based on their ability to predict model performance metrics generated with three distinct sets of features. The null hypothesis, that must be evaluated based on the Friedman

TABLE III: Descriptive Statistics: Different Word Embedding

	Min	Max	Mean	Q2	Q1	Q3
Accuracy						
TFIDF	62.38	94.24	84.52	84.62	82.56	89.47
CBOW	59.68	87.57	77.80	77.77	74.74	82.88
SKG	73.17	91.23	82.34	80.56	77.76	87.19
GLOVE	74.92	90.58	82.42	81.51	78.57	85.76
W2V	73.65	91.49	83.50	82.62	79.92	88.00
Precision						
TFIDF	0.62	0.95	0.85	0.85	0.83	0.90
CBOW	0.60	0.92	0.79	0.78	0.75	0.86
SKG	0.71	0.93	0.83	0.83	0.78	0.89
GLOVE	0.71	0.93	0.83	0.84	0.79	0.88
W2V	0.74	0.94	0.84	0.84	0.79	0.90
Recall						
TFIDF	0.48	0.98	0.85	0.85	0.82	0.92
CBOW	0.46	0.94	0.79	0.80	0.75	0.85
SKG	0.65	0.97	0.83	0.83	0.79	0.89
GLOVE	0.69	0.91	0.83	0.84	0.80	0.87
W2V	0.71	0.95	0.84	0.84	0.80	0.88
AUC						
TFIDF	0.71	0.98	0.91	0.92	0.90	0.96
CBOW	0.64	0.94	0.85	0.86	0.83	0.89
SKG	0.80	0.96	0.89	0.89	0.86	0.93
GLOVE	0.84	0.96	0.89	0.89	0.87	0.93
W2V	0.81	0.97	0.91	0.91	0.88	0.94

TABLE IV: Friedman test : Mean Rank

	Accuracy	Precision	Recall	AUC
DL				
DL1	7.33	6.28	5.17	7.28
DL2	3.53	3.94	4.39	3.25
DL3	3.69	4.02	4.28	3.16
DL4	3.83	4.13	4.35	3.65
DL5	3.84	4.13	4.13	3.65
DL6	4.45	4.43	4.44	4.23
DL7	4.51	4.33	4.88	5.22
DL8	4.83	4.75	4.36	5.58
	P<0.05	P<0.05	P<0.05	P<0.05
Word-Embedding				
TFIDF	1.95	2.18	2.01	1.88
CBOW	4.82	4.51	4.09	4.88
SKG	2.96	3.02	2.98	3.05
GLOVE	3.02	2.99	3.11	3.24
W2V	2.25	2.30	2.81	1.96
	P<0.05	P<0.05	P<0.05	P<0.05
Feature Sets				
AF	1.74	1.71	1.83	1.81
RST	1.58	1.76	1.74	1.47
PCA	2.68	2.53	2.43	2.73
	P<0.05	P<0.05	P<0.05	P<0.05
OD and SMOTE				
OD	1.74	1.64	1.47	1.77
SMOTE	1.26	1.36	1.53	1.23
	P<0.05	P<0.05	P<0.05	P<0.05

test, is that "the requirements classification models developed using different feature sets do not have a significant difference in their prediction capacity." With two degrees of freedom and a significance threshold of $\alpha=0.05$, the Friedman test was conducted. Table IV shows the mean ranks of the three feature sets. The mean ranks of the Friedman test can be used to differentiate between different feature selection techniques. Lower mean ranks imply better achievement as compared to others. The models trained with the set of RST features had the lowest mean rank (1.47), followed by those trained with the original set of features (1.81), and finally PCA (2.73). The mean ranks show that models perform better when

TABLE V: Descriptive Statistics: Different Sets of Features

	Min	Max	Mean	Q2	Q1	Q3
Accuracy						
AF	66.51	94.24	83.14	83.20	78.91	87.17
RST	75.56	92.41	83.59	83.28	79.92	87.43
PCA	59.68	91.23	79.62	78.64	75.84	84.64
Precision						
AF	0.66	0.95	0.84	0.84	0.79	0.89
RST	0.71	0.93	0.84	0.85	0.79	0.89
PCA	0.60	0.94	0.80	0.80	0.76	0.88
Recall						
AF	0.46	0.98	0.84	0.84	0.80	0.88
RST	0.70	0.98	0.84	0.84	0.80	0.89
PCA	0.48	0.97	0.81	0.82	0.77	0.87
AUC						
AF	0.77	0.98	0.90	0.90	0.87	0.93
RST	0.84	0.97	0.91	0.91	0.88	0.93
PCA	0.64	0.96	0.87	0.86	0.84	0.92

RST features are included, whereas dimensionality reduction approaches like PCA just regress the models' performance.

C. RQ3: For what structure of the deep learning classifiers do the software requirements classification models achieve the best results?

The study used eight different Deep Learning classifiers to categorize the software requirements. These classifiers are employed in conjunction with various word-embedding techniques and feature selection techniques. The DL binary classification models include models with different layer sizes, and layer types. Hyperparameters such as optimizer, activation function for specific layers, loss function, etc. are kept consistent across different models.

1) Box-plot: Classification Techniques: Figure 6 depicts the accuracy, precision, recall, and AUC statistics for the

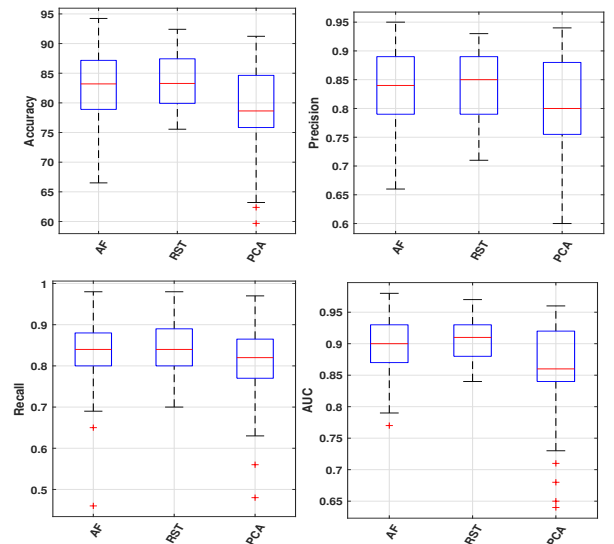


Fig. 5: Performance Box-Plot Diagram: Performance of Different Sets of Features

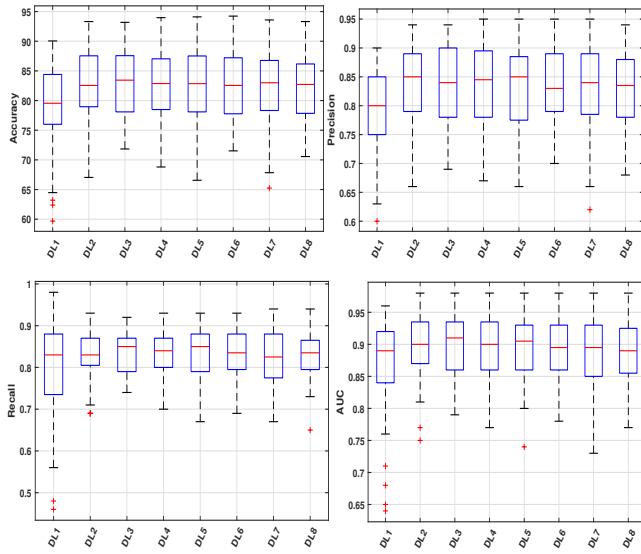


Fig. 6: Performance Box-Plot Diagram: Different Variants of Deep Learning

different classifiers used, including Mean, Median, Min, Max, Q1, and Q3. With a mean AUC of 0.9, the models trained with the DL2, DL3, DL4, and DL5 exhibited relatively similar performance and surpassed the others. The DL1 classifier performed the worst in comparison to other classifiers, with a mean AUC of 0.86. The DL3 classifier produces models with a maximum AUC of 0.98, a minimum of 0.79, a Q1 AUC of 0.86, and a Q3 AUC of 0.94. A box plot is insufficient to discriminate between the DL2, DL3, DL4, and DL5 classifiers.

2) *Friedman Test: Classification Techniques:* The Friedman test is also performed on the performance metrics of the various classifiers in order to statistically compare the models' performance and help differentiate the models where box-plot failed. The goal of the test is to see if the null hypothesis is correct. The null hypothesis for this test is that "the requirements classification models developed utilizing the different classifiers do not have a significant variation in their prediction abilities." With 7 degrees of freedom and a significance level of $\alpha=0.05$, the Friedman test was performed. The mean rank of various classifiers after the Friedman test is shown in Table IV. Table IV shows that DL3 has the lowest mean rank of 3.16, followed by DL2 with 3.25, DL4, and DL5 with 3.65. DL1 and DL8 performed significantly worse with mean ranks of 7.28, and 5.58, respectively.

D. RQ4: How does the application of class balancing technique through oversampling affect the performance of the models?

Based on the data utilized to train the models, there are two types of models presented in this study. One dataset contains imbalanced classes, while the class-balanced dataset is used to train the other set of models.

1) *Box-plot: SMOTE:* Figure 7 presents a visual representation of the predictive ability of models trained on a balanced

TABLE VI: Descriptive Statistics: Different Variants of DL

	Min	Max	Mean	Q2	Q1	Q3
Accuracy						
DL1	59.68	90.05	79.06	79.55	76.00	84.40
DL2	67.04	93.32	82.79	82.55	78.96	87.55
DL3	71.84	93.19	82.93	83.44	78.09	87.57
DL4	68.80	93.98	82.67	82.87	78.49	87.04
DL5	66.56	94.11	82.70	82.85	78.09	87.52
DL6	71.52	94.24	82.54	82.55	77.77	87.22
DL7	65.24	93.59	82.02	83.00	78.33	86.77
DL8	70.56	93.32	82.22	82.72	77.84	86.17
Precision						
DL1	0.60	0.90	0.80	0.80	0.75	0.85
DL2	0.66	0.94	0.84	0.85	0.79	0.89
DL3	0.69	0.94	0.84	0.84	0.78	0.90
DL4	0.67	0.95	0.84	0.85	0.78	0.90
DL5	0.66	0.95	0.83	0.85	0.78	0.89
DL6	0.70	0.95	0.83	0.83	0.79	0.89
DL7	0.62	0.95	0.83	0.84	0.79	0.89
DL8	0.68	0.94	0.83	0.84	0.78	0.88
Recall						
DL1	0.46	0.98	0.81	0.83	0.74	0.88
DL2	0.69	0.93	0.83	0.83	0.81	0.87
DL3	0.74	0.92	0.84	0.85	0.79	0.87
DL4	0.70	0.93	0.83	0.84	0.80	0.87
DL5	0.67	0.93	0.84	0.85	0.79	0.88
DL6	0.69	0.93	0.83	0.84	0.80	0.88
DL7	0.67	0.94	0.83	0.83	0.78	0.88
DL8	0.65	0.94	0.83	0.84	0.80	0.87
AUC						
DL1	0.64	0.96	0.86	0.89	0.84	0.92
DL2	0.75	0.98	0.90	0.90	0.87	0.94
DL3	0.79	0.98	0.90	0.91	0.86	0.94
DL4	0.77	0.98	0.90	0.90	0.86	0.94
DL5	0.74	0.98	0.90	0.91	0.86	0.93
DL6	0.78	0.98	0.90	0.90	0.86	0.93
DL7	0.73	0.98	0.89	0.90	0.85	0.93
DL8	0.77	0.98	0.89	0.89	0.86	0.93

dataset versus models learned on an imbalanced dataset. In most box plot measures, models trained with SMOTE outperformed models trained on the original dataset. SMOTE-trained models had a mean AUC of 0.9, a maximum of 0.98, a minimum of 0.64, and a Q3 of 0.95.

TABLE VII: Descriptive Statistics: OD and SMOTE

	Min	Max	Mean	Q2	Q1	Q3
Accuracy						
OD	63.20	91.36	81.20	81.84	77.76	84.80
SMOTE	59.68	94.24	83.03	83.50	78.18	88.68
Precision						
OD	0.63	0.94	0.82	0.83	0.78	0.88
SMOTE	0.60	0.95	0.84	0.84	0.78	0.90
Recall						
OD	0.67	0.98	0.83	0.84	0.79	0.88
SMOTE	0.46	0.93	0.83	0.84	0.79	0.87
AUC						
OD	0.65	0.96	0.88	0.89	0.86	0.92
SMOTE	0.64	0.98	0.90	0.91	0.86	0.95

2) *Friedman Test: SMOTE:* To evaluate the performance of the two sets of models, the Friedman test is applied to their performance measures on class-balanced and imbalanced datasets. The goal of this test is to accept or reject the null hypothesis, which states that "there is no substantial difference in performance between models trained with balanced or imbalanced classes." With a significance threshold of $\alpha = 0.05$

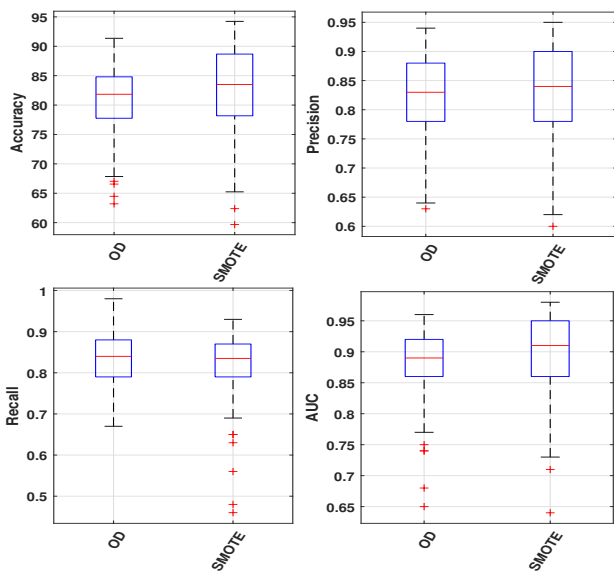


Fig. 7: Performance Box-Plot Diagram: Performance of Original Data and SMOTE

and degrees of freedom of 1, the Friedman test was performed. The models trained on the class-balanced dataset have the lower mean rank of 1.23, while those trained on the imbalanced dataset have the higher mean rank of 1.77.

VII. CONCLUSION

The Deep Learning classifiers used in this work, in conjunction with the word embedding, feature selection, and class balancing techniques have produced a very high accuracy for software requirements classification. The best models can be deployed in industry to do away with the manual classification, which is ailed by the problem of inconsistencies between client and software engineer terminologies [13]. Industry can benefit from the lower costs, and higher efficiency. The key conclusions that we arrived at were as follows:

- Models that utilized features extracted from TF-IDF word embedding performed considerably better than other models.
- Out of the eight DL classifiers used, the models trained with the DL3 classifier had the best results, with the DL2 and DL4 classifiers performing similarly.
- The Rank Sum test-selected features outperformed any other group of characteristics utilized in this work.
- SMOTE-based class balancing improved the performance of requirements categorization models.

According to the results of this study, DL classifiers are critical for more accurate software requirement classification. Future

research can extend this work by classifying the requirements into the subcategories of functional and non-functional requirements. Researchers can also focus on other parts of the software development pipeline, and use the models developed in this work for the requirements classification step.

VIII. ACKNOWLEDGEMENTS

This research is funded by TestAIng Solutions Pvt. Ltd.

REFERENCES

- [1] M. V. Mäntylä, F. Calefato, and M. Claes, "Natural language or not (nlon): A package for software engineering text analysis pipeline," in *Proceedings of the 15th International Conference on Mining Software Repositories*, ser. MSR '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 387–391. [Online]. Available: <https://doi.org/10.1145/3196398.3196444>
- [2] J. Cleland-Huang, R. Settini, X. Zou, and P. Solc, "Automated classification of non-functional requirements," *Requirements engineering*, vol. 12, no. 2, pp. 103–120, 2007.
- [3] M. H. Osman and M. F. Zaharin, "Ambiguous software requirement specification detection: An automated approach," in *2018 IEEE/ACM 5th International Workshop on Requirements Engineering and Testing (RET)*, 2018, pp. 33–40.
- [4] E. Knauss, D. Damian, G. Poo-Caamaño, and J. Cleland-Huang, "Detecting and classifying patterns of requirements clarifications," in *2012 20th IEEE International Requirements Engineering Conference (RE)*, 2012, pp. 251–260.
- [5] R. Navarro-Almanza, R. Juarez-Ramirez, and G. Licea, "Towards supporting software engineering using deep learning: A case of software requirements classification," in *2017 5th International Conference in Software Engineering Research and Innovation (CONISOFT)*, 2017, pp. 116–120.
- [6] A. Araujo and R. Marcacini, "Hierarchical cluster labeling of software requirements using contextual word embeddings," in *Brazilian Symposium on Software Engineering*, 2021, pp. 297–302.
- [7] D. Ott, "Automatic requirement categorization of large natural language specifications at mercedes-benz for review improvements," in *International Working Conference on Requirements Engineering: Foundation for Software Quality*. Springer, 2013, pp. 50–64.
- [8] C. Baker, L. Deng, S. Chakraborty, and J. Dehlinger, "Automatic multi-class non-functional software requirements classification using neural networks," in *2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC)*, vol. 2, 2019, pp. 610–615.
- [9] N. Rahimi, F. Eassa, and L. Elrefaie, "An ensemble machine learning technique for functional requirement classification," *symmetry*, vol. 12, no. 10, p. 1601, 2020.
- [10] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.
- [11] S. Tiun, U. Mokhtar, S. Bakar, and S. Saad, "Classification of functional and non-functional requirement in software requirement using word2vec and fast text," in *journal of Physics: conference series*, vol. 1529, no. 4. IOP Publishing, 2020, p. 042077.
- [12] J. Hassine, R. Dssouli, and J. Rilling, "Applying reduction techniques to software functional requirement specifications," in *System Analysis and Modeling*, D. Amyot and A. W. Williams, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 138–153.
- [13] C. P. Guevara-Vega, E. D. Guzmán-Chamorro, V. A. Guevara-Vega, A. V. B. Andrade, and J. A. Quiña-Mera, "Functional requirement management automation and the impact on software projects: case study in ecuador," in *International Conference on Information Technology & Systems*. Springer, 2019, pp. 317–324.

7th Workshop on Model Driven Approaches in System Development

FOR many years, various approaches in system design and implementation differentiate between the specification of the system and its implementation on a particular platform. People in software industry have been using models for a precise description of systems at the appropriate abstraction level without unnecessary details. Model-Driven (MD) approaches to the system development increase the importance and power of models by shifting the focus from programming to modeling activities. Models may be used as primary artifacts in constructing software, which means that software components are generated from models. Software development tools need to automate as many as possible tasks of model construction and transformation requiring the smallest amount of human interaction.

A goal of the proposed workshop is to bring together people working on MD approaches, techniques and tools, as well as Domain Specific Modeling (DSM) and Domain Specific Languages (DSL) and applying them in the requirements engineering, information system and application development, databases, and related areas, so that they can exchange their experience, create new ideas, evaluate and improve MD approaches and spread its use. The intention is to target an interdisciplinary nature of MD approaches in software engineering, as well as research topics expressed by but not limited to acronyms such as Model Driven Software Engineering (MDSE), Model Driven Development (MDD), Domain Specific Modeling (DSM), and OMG's Model Driven Architecture (MDA).

1st Workshop on MDASD was organized in the scope of ADBIS 2010 Conference, held in Novi Sad, Serbia. From 2012, MDASD becomes a regular bi-annual FedCSIS event.

TOPICS

- MD Approaches in System Design and Implementation – Problems and Issues
- MD Approaches in Software Process Models, Software Quality and Standards
- MD Approaches in Databases, Information Systems, Embedded and Real-Time Systems
- Metamodeling, Modeling and Specification Languages, Model Transformation Languages
- Model-to-Model, Model-to-Text, and Model-to-Code Transformations in Software Process
- Transformation Techniques and Tools
- Domain Specific Languages (DSL) and Domain Specific Modeling (DSM) in System Specification and Development
- Design of Metamodeling and Modeling Languages and Tools
- MD Approaches in Requirements Engineering, Document Engineering and Business Process Modeling
- MD Approaches in System Reengineering and Reverse Engineering
- MD Approaches in HCI and UX Design, GIS Development, and Cyber-Physical Systems
- Low-Code and No-Code software development – research, experiences and challenges
- Model Based Software Verification
- Artificial Intelligence (AI) for MD and MD for AI-based system development
- Theoretical and Mathematical Foundations of MD Approaches
- Multi-View, Multi-Paradigm and Blended Modeling
- Organizational and Human Factors, Skills, and Qualifications for MD Approaches
- Teaching MD Approaches in Academic and Industrial Environments
- MD Applications, Industry Experience, Case Studies, relationship with IoT and Industry 4.0

There is a possibility of selecting extended versions of the best papers presented during the conference for further procedure in the journals: ComSIS, ISI IF(2020) = 1.167, and COLA, ISI IF(2020) = 1.271.

TECHNICAL SESSION CHAIRS

- **Luković, Ivan**, University of Novi Sad, Serbia

STEERING COMMITTEE

- **Gray, Jeff**, University of Alabama, United States
- **Mernik, Marjan**, University of Maribor, Slovenia
- **Ristić, Sonja**, University of Novi Sad, Faculty of Technical Sciences, Serbia
- **Tolvanen, Juha-Pekka**, MetaCase, Finland

PROGRAM COMMITTEE

- **Amaral, Vasco**, NOVA University Lisbon, Portugal
- **Brdjanin, Drazan**, University of Banja Luka, Faculty of Electrical Engineering, Bosnia and Herzegovina
- **Bryant, Barrett**, University of North Texas, USA
- **Budimac, Zoran**, Faculty of Sciences, University of Novi Sad, Serbia
- **Chen, Haiming**, Chinese Academy of Sciences, China
- **Erradi, Mohammed**, ENSIAS Rabat, Morocco

- **Fertalj, Krešimir**, Faculty of EE and Computing, University of Zagreb, Croatia
- **Haerting, Ralf**, Hochschule Aalen, Germany
- **Ivanovic, Mirjana**, Faculty of Sciences, University of Novi Sad, Serbia
- **Janousek, Jan**, Czech Technical University Prague, Czech Republic
- **Karagiannis, Dimitris**, University of Vienna, Austria
- **Kardaş, Geylani**, Ege University, Turkey
- **Kordić, Slavica**, Faculty of Technical Sciences, University of Novi Sad, Serbia
- **Kosar, Tomaz**, University of Maribor, Slovenia
- **Krdzavac, Nenad**, University of Belgrade, Serbia
- **Liu, Shih-Hsi**, California State University, Fresno, USA
- **Macos, Dragan**, Beuth Hochschule für Technik, Germany
- **Melo-de-Sousa, Simão**, Universidade da Beira Interior, Portugal
- **Milosavljevic, Gordana**, Faculty of Technical Sciences, University of Novi Sad, Serbia
- **Özkaya, Mert**, Istanbul Kemerburgaz University, Turkey
- **Porubán, Jaroslav**, Technical University of Košice, Slovakia
- **Rangel Henriques, Pedro**, University of Minho, Portugal
- **Selic, Bran**, Malina Software Corp., Canada
- **Sierra, Jose Luis**, Universidad Complutense de Madrid, Spain
- **Slivnik, Bostjan**, University of Ljubljana, Slovenia
- **Tavakoli Kolagari, Ramin**, Nuremberg Institute of Technology, Germany
- **Varanda Pereira, Maria João**, Instituto Politécnico de Bragança, Portugal
- **Vescoukis, Vassilios**, National Technical University of Athens, Greece
- **Wimmer, Manuel**, JKU Linz, Austria

Model-Based System Engineering Adoption in the Vehicular Systems Domain

Henrik Gustavsson¹, Eduard Paul Enoiu¹, Jan Carlson¹

¹School of Innovation, Design and Engineering, Mälardalen University, Västerås, Sweden

Abstract—As systems continue to increase in complexity, some companies have turned to Model-Based Systems Engineering (MBSE) to address different challenges such as requirement complexity, consistency, traceability, and quality assurance during system development. Consequently, to foster the adoption of MBSE, practitioners need to understand what factors are impeding or promoting success in applying such a method in their existing processes and infrastructure. While many of the existing studies on the adoption of MBSE in specific contexts focus on its applicability, it is unclear what attributes foster a successful adoption of MBSE and what targets the companies are setting. Consequently, practitioners need to understand what adoption strategies are applicable. To shed more light on this topic, we conducted semi-structured interviews with 12 professionals working in the vehicular domain with roles in several MBSE adoption projects. The aim is to investigate their experiences, reasons, targets, and promoting and impeding factors. The obtained data was synthesized using thematic analysis. This study suggests that the reasons for MBSE adoption relate to two main themes: better management of complex engineering tasks and communication between different actors. Furthermore, engagement, activeness and access to expert knowledge are indicated as factors promoting MBSE adoption success, while the lack of MBSE knowledge is an impeding factor for successful adoption.

I. INTRODUCTION

MODEL-Based Systems Engineering (MBSE) [1] is an approach devised to take care of issues frequently encountered in traditional document-based system engineering (DBSE). In MBSE practice the systems engineering team performs its engineering life cycle activities in a modelling tool, using a dedicated semi-formal modelling language and applying a modelling method, to construct one primary systems engineering artifact — a system model which is inherently coherent and consistent [2]. The great appeal of MBSE is that it, when practiced correctly, promises a return on investment (ROI) that appears late in the Systems Engineering life cycle due to reduced costs in change management. However sometimes stakeholders also incorrectly assume that MBSE makes every systems engineering activity easier and cheaper [2].

Douglass [3] suggests that adoption of MBSE is a challenging change process featuring four partly overlapping phases - assessment, planning, piloting/early adoption, and deployment, where the level of success in each phase depends strongly on the quality of the work that has been done in the previous phase. The empirical results of a study by Rogers and Mitchell suggest that the investment cost for transitioning to MBSE could be considerable and that the adopting organization might have to display patience regarding the ROI [4].

Due to the challenges and the complexity, studying MBSE adoption cases in an industrial context is of high value for other industrial organizations aiming to adopt MBSE. Especially understanding the intentions and related experiences when organizations are setting out for MBSE adoption will provide valuable knowledge upon which future adoptions can be based.

This paper presents an exploratory interview study searching for MBSE adoption purposes, targets and factors promoting or impeding success. The participants in the study were interviewed about their individual experiences related to the above. Three research questions were defined in this study:

- RQ1. What are the primary reasons for and targets in MBSE adoption?
- RQ2. What factors are promoting success in MBSE adoption?
- RQ3. What factors are impeding success in MBSE adoption?

The study was conducted by interviewing practitioners and other stakeholders and applying thematic analysis to the data collected in the interviews to identify themes for each of the research questions. Reasons for MBSE adoption were grouped into two different themes – manage complex engineering tasks better and achieve effective communication and collaboration. Factors promoting MBSE adoption success were also grouped into two different themes – activeness and engagement and access to MBSE expert knowledge. The factors impeding MBSE adoption success were collected under one theme – insufficient MBSE knowledge.

A. Background and Related Work

Douglass [3] suggests that the adoption of a new approach encompassing a new language, such as MBSE with SysML, is characterized by four overlapping phases - assessment, planning, piloting/early adoption, and deployment. In this section, we survey some work related to these MBSE adoption phases.

1) *Reasons and targets*: Mitchell et al. [4], [5] have performed empirical case studies on the transitioning to MBSE in a system-of-systems product family organization. The primary purposes of the transition were to keep up with the increasing workload, increase automation in the systems engineering workflow, eliminate duplicate data, enhance manual quality assurance, enhance change impact analysis, achieve the automated generation of an interface description language, improve data integrity, and reduce the cost of quality assurance. Carroll

et al. [6] have found that the arguments justify MBSE adoption that will enable improvement of engineering efficiency and prevention of costly rework. In addition, Chaudron et al. [7] have synthesized empirical evidence regarding the effectiveness of UML modeling in software development. The study concludes that the two ultimate benefits of UML modeling are improved quality and higher productivity, both of which stem from the direct benefits which UML modeling brings to the developer and the team – UML modeling stimulates the developer to think harder and hence better understand the problem domain and the solution space.

2) *Factors promoting and impeding success:* Mitchell presents some lessons learned from MBSE introduction [5] — there is a big learning curve to consider, the strive for efficiency requires re-engineering the business process, and if consistency is important, one has to manage human resistance.

Amorim et al. [8] have performed a study to find strategies and best practices for MBSE adoption in the embedded systems industry. They conclude that the advantages of MBSE shall be made clear to the adopting team, the organization shall start the adoption on a small scale, and all engineers should get at least basic MBSE training. Hallqvist et al. [9] have done an empirical study on the introduction of MBSE by using systems engineering principles. In their study, they presented several lessons learned, namely to keep the focus on the purpose, start small while thinking big, address all stakeholders, involve people that have gone through a similar process before, have leadership present who understands people, have a communication plan, and consider using prototyping for validating changes. Madni and Purohit [10] proposed a framework for analyzing investments and potential gains when implementing MBSE. Their results support the view that MBSE requires an upfront investment, with gains showing up in later system life cycle stages. They mention several gains of MBSE such as early defect detection, reuse, product line definition, risk reduction, improved communication, usage in the supply chain and standards conformance that are important. Suryadevara et al. [11] imply that significant investment, a considerable learning effort and attainment of good tool interoperability are components required for success. Selberg et al. [12] have studied MBSE adoption in a company, based on which they give recommendations for adopting MBSE. Their recommendations are to clearly define the purpose of the adoption, assemble a core team, plan for the changes, allow sufficient time, and provide sufficient training to all stakeholders.

What is scarce in the current work is granularity and visibility of data associated with parties and phases in MBSE adoption, including parties who have little or no direct contact with the model and including the phases from assessment to deployment. More empirical knowledge is needed on these facets of adoption.

II. METHOD

This study was conducted through semi-structured interviews, following Strandberg [13] as the primary interview guidelines. The interviews were transcribed and then analyzed

using Braun and Clarke's guidelines for thematic analysis [14]. For more information on the method used we refer the reader to the extended technical report of Gustavsson et al. [15].

A. Planning

To plan and keep track of the work, an interview survey plan was written according to the guidelines by Linåker et al. [16]. As the work on the interview study progressed, the plan was also used to record changes. First, a raw survey instrument with 22 questions was created. In a workshop amongst the authors, we refined it and organized it into initial question groups (topics). Then, in a series of iterations, we created and refined a survey instrument. Each interview was planned with a start session where we would explain the purpose and motivation as well as the interview process. The first topic in the instrument focused on the interviewee (e.g., background, work experience and knowledge related to MBSE). In contrast, the last topic was related to successes, setbacks, and other experiences during the adoption and deployment of MBSE.

B. Interviews

We recruited a convenience sample of individuals affiliated with an organization in the embedded system domain. Using a stratified design to ensure experience and specialization diversity related to MBSE, we selected individuals from the following groups: managers, modelers, and model users. The interviewees were selected from a diverse set of MBSE adoption projects inside the company using a convenience sample based on our contacts in the company. In total, we recorded about 11 hours of audio material.

C. Transcription and Thematic Analysis

The interviews were transcribed using reflective journalism transcription [17] into 109 pages of text. During the transcription, we ensured the anonymization of the transcript. Text coding, was done in the following way: One interview was independently coded by all three authors and the three results were compared, discussed and adjusted to build consensus on the procedure. The remaining interviews were coded by two authors independently, and then discussed in a joint workshop to align the alternatives and agree upon a final coding. For the thematic analysis, we used the Braun and Clarke method [14] and the Halcomb data management steps [17]. A preliminary thematic analysis was done by the first author to elicit a first version of the main themes and then thoroughly reviewed by the other authors based on both audiotapes and interview notes. Next, we iterated on the sets of themes. This activity ended with a workshop with all authors where the final set of themes was agreed upon.

III. RESULTS

We start this section with an overview of the organization and interviewees (also outlined in Table I). Then, the main part of this section covers the thematic analysis results, the overall thematic map in Figure 1, and the answers to our research questions. For more details on the results we refer the reader to the extended technical report of Gustavsson et al. [15].

TABLE I: Interview participants including roles in adoption, expertise domain, self rated MBSE knowledge before and after adoption, and the adoption cases in which they were involved

Interviewee	Role in MBSE adoption			Expertise domain	MBSE knowledge		Cases					
	Modeller	Model user	Team manager		Before	After	1	2	3	4	5	
#1	x	x	x	Mathematical modelling, 5 years	1	3	x					
#2			x	Product functions, 27 years	2	5		x				
#3		x		Software design, 6 years	2	2	x					
#4	x		x	Product functions and project management, 8 years	2-3	3			x			
#5		x		Safety related embedded systems engineering, 8 years	2	2	x					
#6	x	x		Safety software architect, 11 years	1	2-3	x					
#7		x		Software development, 18 years	1	4	x					
#8		x		Software development and project management, 20-25 years	1-2	2-3	x					
#9	x			Control and systems engineering, 13 years	1	4		x				
#10	x	x		Subsystems functional design and control system functions, 7 years	0	3	x					
#11	x		x	Mechanical engineering, 30 years	1	2						x
#12	x		x	System design, 15 years	4	3	x				x	

A. Context, Organization and Interviewees

We conducted semi-structured interviews with twelve individuals from the same organization. The organization develops embedded systems in the domain of safety-critical vehicular systems and has more than 35 000 employees over many sites. The company is undergoing a transition to MBSE and has started different adoption initiatives in different sites.

Table I provides some basic information about the interviewees, including their roles and expertise areas. The interviewees were also asked to self-assess their MBSE knowledge before and after the adoption. In total, the interviews covered the experience of the interviewees from five different MBSE adoption cases: Cases 1-4 (Adoptions in the deployment phase) and Case 5 (Adoption in the early adopter/piloting phase). Six interviewees had been in more than one role related to MBSE adoption, and one interviewee had been involved in two different cases. The participants can be categorized in the following roles¹: modellers (seven interviewees), model users (seven interviewees), and team managers (five interviewees). The interviewees have between 5 and 30 years of experience, with an average of 14.4 years. Seven had at least ten years of work experience. Typical domains of expertise for our interviewees were system design and software development, mathematical modelling, mechanical engineering, software architecture, project management and embedded system engineering. Related to their MBSE understanding, most participants rated themselves as having relatively low knowledge before adoption. According to this self-evaluation, most participants have seen their MBSE knowledge improve during their work in each case.

B. RQ1: Primary reasons for adoption and targets in adoption

In this part of the study, we identified two themes. Both themes were related to primary reasons for adoption: to manage complex engineering tasks in a better way and to achieve effective communication and collaboration. Unfortunately, the interview data did not yield any themes on quantified targets.

1) *Theme: Manage complex engineering tasks better:* MBSE was seen as an enabler for managing complex technical

engineering tasks more efficiently and effectively than traditional system engineering methods. The notion of complexity in this context seems to be related to the nature of the technical challenge involved, the sheer size of the task or a combination of the two. An example of such an area was the work related to subsystem interfaces where the model and the modelling tool were considered to provide a better environment: “*MBSE makes interface management very accurate.*” Another area where MBSE was considered to provide an attractive capability was requirements verification: “*The big selling point was left shifting verification of the requirements using simulation.*”. Other areas include system testing, change impact analysis and propagation analysis, product homologation, reuse of design solutions, and software standardization.

2) *Theme: Achieve effective communication and collaboration:* MBSE was regarded as a means to facilitate and enable effective communication and collaboration in a way that is not possible without MBSE. A vision presented by a participant was that the opportunity to represent design in a uniform way in MBSE should be exploited such that it facilitates the communication across the entire MBSE organization, as well as with external stakeholders: “*I think the major objective of MBSE shall be to provide a uniform way of representing the design that we are doing... If you want all the regions to understand what others are doing, and with suppliers and things like that.*”

C. RQ2: Factors promoting success in Adoption

In this part of the study, we identified the following two themes: (i) activeness and engagement as well as (ii) access to expert knowledge.

1) *Theme: Activeness and engagement:* When there were partakers in the team who were active, engaged and persistent, this was associated with MBSE success. The interviews gave observations of managers and model users displaying such qualities. Most of the observations happened when the adoption or deployment was hard going and certain team colleagues were showing a tendency to falter. It was also observed that certain valuable features and instruments for promoting success materialized due to the activeness of the management team. One interviewee stated that the model users being engaged in the modelling tasks had a positive effect on the quality of the requirements derived from the model: “*We*

¹We note here that some participants had overlapping roles as modellers, model users and/or team managers.

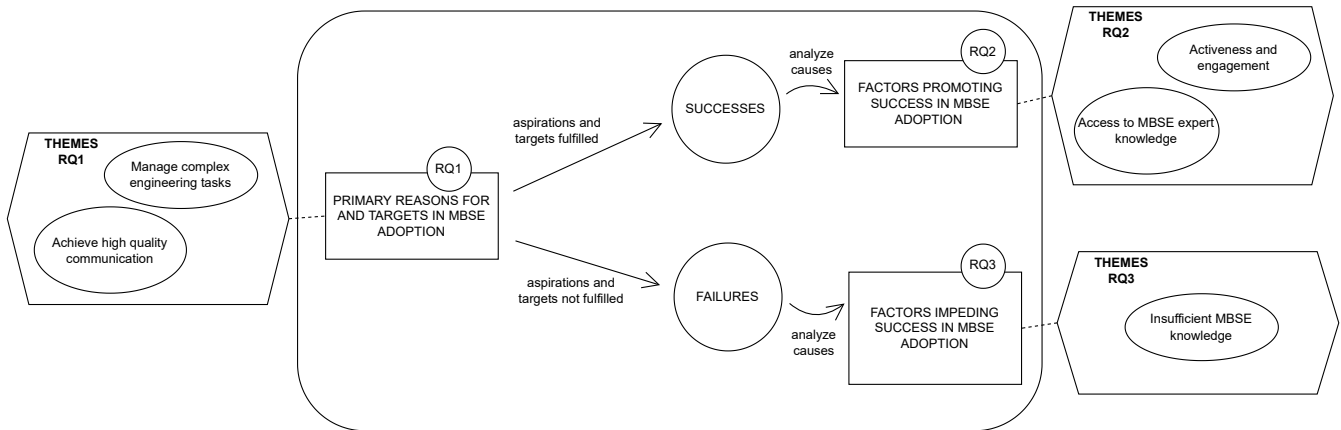


Fig. 1: Overview of the themes and their relation to the research questions.

managed to get the modellers to keep the models updated, then they were thinking a bit extra on their requirements when they were modeling.”

One participant observed that their company management was active in giving their full support to the MBSE adoption including the deployment. This support helped manage resistance and questioning of MBSE in the organization. The interviewee also linked the management support to the institutionalization of an MBSE core team (i.e., expert knowledge) and the accessibility to the model for the whole team. As the management of resistance, the MBSE core team and the team’s model accessibility contributed to the long-term success of the deployment. Related to the successes experienced in the adoption, another participant concluded that these were very much dependent on the adoption team’s engagement and their supporting stakeholders in the initial phase of the adoption. These adoption team members had taken the initiative themselves to learn the subject and their stakeholders had also been very active when soliciting input data.

2) *Theme: Access to MBSE expert knowledge:* A factor behind the MBSE’s success was found to be related to present and readily accessible expertise. The instances showing this in the interviews are: (i) the presence of an MBSE core team that assumed a clear long-term role as MBSE process owner, (ii) the provisioning of continuous team support, and (iii) being attentive and adapting to the needs of the MBSE team. Establishing an MBSE expert inside the modelling team in the very early stage of adoption was also perceived as a factor promoting success. For example, one interviewee had an experience where the expert had defined the MBSE process for the team. Another participant had an experience where the expert had acted as a sounding board for the adopters. The expert had even enabled a doubtful adopter to turn into an enthusiast in this role.

D. RQ3: Factors impeding success in Adoption

In this part of the study, we identified one central theme: insufficient MBSE knowledge.

1) *Theme: Insufficient MBSE Knowledge:* A factor that impedes MBSE adoption success is a lack of knowledge of MBSE. This theme encompasses cognizance on different levels, in diverse areas and among various parties in the MBSE adoption. It appeared in situations when people were, in various ways, dependent on a particular party to make progress in the MBSE adoption. Participants concluded that this involved party had a knowledge gap preventing progress.

An example of such missing knowledge was related to SysML and shown among engineers. While the early phase of SysML adoption among engineers went well, some time into the adoption, an apparent threshold in the overall progress was encountered regarding important concepts in the SysML syntax and semantics. Examples of concepts that caused the adopter’s various degrees of difficulty were the two distinct kinds of flows on activity diagrams, ports, and the internals of blocks. For example, one participant mentioned the following: “When we are getting to more complex things, that is, when we are getting to ports, various types of ports, exactly what they mean and what they do, it is getting more difficult.” There were observations about a weak cognizance of MBSE among people who were not in direct contact with the model, e.g. people in management. When people in management and other stakeholders were approached by adopters regarding issues that the adopters could not resolve among themselves, flaws in the cognizance of MBSE impeded the possibility to support the adopters or act as sounding boards. As the adopters did usually not have the resources needed to resolve the problems this could cause problems to remain unresolved. Among the engineers, the lack of MBSE cognizance can make them nurture expectations that the deliverables of the modelling team will be definitive and that all subsequent collaboration with that team will be superfluous: “People expect that we are going to provide them with requirements and everything will be complete and they can leave from there and they will not have to talk to us anymore.” Insufficient MBSE knowledge could also lead to people comparing model diagrams to other artifacts they could relate to, such as Visio diagrams, whereby

they were resisting and questioning the change. It was difficult to bridge this gap by means of argumentation: “*They just think MBSE is just like Microsoft Paint, you know you draw some pictures, it is just like Visio or something*”

IV. DISCUSSION

For research, the findings in this paper are essential as they bring an industrial experience of MBSE adoption and deployment from industrial practice into academia. By understanding that this is not an insignificant process, additional research is possible. Other researchers could add on and revisit MBSE adoption and deployment in other contexts and possibly investigate the results of this study.

Based on our findings, organizations in the vehicular systems domain adopting and deploying MBSE may want to foster activeness, engagement, and access to MBSE expert knowledge in their teams. In addition, companies should pay special attention to the insufficient MBSE knowledge, especially during the adoption phase. There is relatively little data that supports the research question on quantified targets in MBSE adoption. However, in the responses to the interview question, there is considerably more data about primary reasons for adoption. A few interviewees also expressed satisfaction related to the personal gains in learning a new method. However, when asked to suggest the reasons they thought were behind the positive details, the answers did not provide clear reasons. More research is needed to understand these personal reasons and human aspects of learning MBSE.

When discussing the concept of impeding factors, our results suggest that this is a rather multifaceted topic. First of all, the interview questions were framed to stimulate the interviewees to describe a logically coherent story about the MBSE adoption and deployment. One idea behind the design of the interviews was to avoid retrospectively invented opinions from the interviewees about successes and failures but instead, encourage them to base these ideas on observations as to whether the adoption reasons and targets were met or not. Once the successes or failures had been recollected, the interviewees were asked to consider what could have been the cause of each case. As it turned out, the interviewees had many different ways of expressing successes, failures, and factors promoting success or causing failure. One reason for this is that when asked to name challenges and setbacks, participants proposed a predefined view that seemed necessary to be included in the adoption to make it successful.

V. CONCLUSIONS

We have conducted an interview study of model-based system engineering adoption and deployment in the vehicular domain. The results presented in this paper are based on semi-structured interviews with twelve practitioners with an average work experience of more than fourteen years and thematic analysis to identify major themes around the reasons, targets, and promoting and impeding factors in model-based

system engineering adoption. We discovered that the primary reasons and targets relate to managing complex engineering tasks in better ways and effective communication. Our results suggest that the main factors promoting success are activeness, engagement and access to expert knowledge. A factor that was shown to impede adoption success is the lack of knowledge on different levels and among different parties. Finally, our results show that more research on the model-based system engineering adoption and deployment is needed and that practitioners need to take these aspects more clearly into account.

VI. ACKNOWLEDGMENT

This work has received funding from H2020 under grant agreements No 871319 and No. 737494, from Vinnova through the SmartDelta and MegaM@Rt2 projects and from KKS through ARRAY Academy.

REFERENCES

- [1] A. M. Madni and M. Sievers, “Model-based systems engineering: Motivation, current status, and research opportunities,” *Systems Engineering*, vol. 21, no. 3, pp. 172–190, 2018.
- [2] L. Delligatti, *SysML distilled: A brief guide to the systems modeling language*. Addison-Wesley, 2013.
- [3] B. P. Douglass, *Agile systems engineering*. Morgan Kaufmann, 2015.
- [4] E. B. Rogers and S. W. Mitchell, “Mbse delivers significant return on investment in evolutionary development of complex sos,” *Systems Engineering*, vol. 24, pp. 385–408, 2021.
- [5] S. W. Mitchell, “Transitioning the swifts program combat system product family from traditional document-centric to model-based systems engineering,” *Systems Engineering*, vol. 17, no. 3, pp. 313–329, 2014.
- [6] E. R. Carroll and R. J. Malins, “Systematic literature review: How is model-based systems engineering justified?” 2016.
- [7] M. Chaudron, W. Heijstek, and A. Nugroho, “How effective is uml modeling?” *Software & Systems Modeling*, vol. 11, no. 4, p. 571, 2012.
- [8] T. Amorim, A. Vogelsang, F. Pudlitz, P. Gersing, and J. Philipps, “Strategies and best practices for model-based systems engineering adoption in embedded systems industry,” in *International Conference on Software Engineering*. IEEE, 2019, pp. 203–212.
- [9] J. Hallqvist and J. Larsson, “Introducing mbse by using systems engineering principles,” in *INCOSE International Symposium*, vol. 26, no. 1. Wiley Online Library, 2016, pp. 512–525.
- [10] A. M. Madni and S. Purohit, “Economic analysis of model-based systems engineering,” *Systems*, vol. 7, no. 1, p. 12, 2019.
- [11] J. Suryadevara and S. Tiwari, “Adopting mbse in construction equipment industry: An experience report,” in *Asia-Pacific Software Engineering Conference (APSEC)*. IEEE, 2018, pp. 1–10.
- [12] O. Selberg and V. Åsberg, “Recommendations for introducing model based systems engineering, master’s thesis in software engineering,” University of Gothenburg, Tech. Rep., 2017.
- [13] P. E. Strandberg, “Ethical interviews in software engineering,” in *2019 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*. IEEE, 2019, pp. 1–11.
- [14] V. Braun and V. Clarke, “Using thematic analysis in psychology,” *Qualitative research in psychology*, vol. 3, no. 2, pp. 77–101, 2006.
- [15] H. Gustavsson, E. P. Enouï, and J. Carlson, “Model-based system engineering adoption in the vehicular systems domain,” MDH-MRTC-344/2022-1-SE, Mälardalen Real-Time Research Centre, Mälardalen University, Tech. Rep., July 2022.
- [16] J. Linåker, S. M. Sulaman, M. Höst, and R. M. de Mello, “Guidelines for conducting surveys in software engineering v. 1.1,” Lund University, Tech. Rep., 2018.
- [17] E. J. Halcomb and P. M. Davidson, “Is verbatim transcription of interview data always necessary?” *Applied nursing research*, vol. 19, no. 1, pp. 38–42, 2006.

An Experimentation Framework for Specification and Verification of Web Services

Szymon Katra
Warsaw University of Technology
Faculty of Electronics
and Information Technology
Nowowiejska Str. 15/19
00-665 Warsaw, Poland
Email: szymon.katra.stud@pw.edu.pl

Wiktor B. Daszczuk
Warsaw University of Technology
Institute of Computer Science
Nowowiejska Str. 15/19
00-665 Warsaw, Poland
Email: wiktor.daszczuk@pw.edu.pl

Denny B. Czejdo
Fayetteville State University
Department of Mathematics and
Computer Science
Fayetteville, NC 28301, USA
Email: bczejdo@uncfsu.edu

Abstract—Designing and implementing Web Services constitutes a large and constantly growing part of the information technology market. Web Services have specific scenarios in which distributed processes and network resources are used. This aspect of services requires integration with the model checkers. This article presents the experimentation framework in which services can be specified and then formally analyzed for deadlock-freedom, achievement of process goals, and similar features. Rybu4WS language enriches the basic *Rybu* language with the ability to use variables in processes, service calls between servers, new structural instructions, and other constructions known to programmers while remaining in line with declarative, mathematical *IMDS* formalism. Additionally, the development environment allows simulation of a counterexample or a witness - obtained as a result of the model checking - in a similar way to traditional debuggers.

I. INTRODUCTION

ARISING number of available Web Services are used for business processes in the modern world. Interactions with other Web Services are key features in creating more complex scenarios and satisfying business needs. An example of interaction between different services might be a travel agency that uses external services to book hotels, flights, and other facilities. Service for booking flights may use another service for processing a payment that communicates with a bank or credit card provider. Such interaction between many services is called *Web Service composition* [1]. From the computer science point of view, Web Service composition is a distributed system concerned with typical problems like deadlocks or lack of termination.

In Warsaw University of Technology, Institute of Computer Science, an experimentation framework for specification and verification of web services composition was developed. It is based on *IMDS* formalism (Integrated Model of Distributed Systems [2]) and *DedAn* [3] tool to model asynchronous distributed systems and verify them automatically. The user does not need to have deep knowledge about verification methods such as temporal logic and model checking.

A distributed system under verification in *IMDS* formalism is defined as a set of actions used to model the behavior. The declarative input of the *DedAn* environment was designed to structure a set of actions by combining them across servers or agents. While *DedAn* input language is sufficient to specify simple distributed systems, modeling more complex cases is challenging due to various technical difficulties described later in this paper.

To overcome the problem of modeling complex distributed systems, a higher-level language *Rybu* was initially developed [4], which simplifies the modeling of the system by some imperative-style elements and data aggregation. To improve the modeling of Web Service composition, *Rybu4WS* language was created, which is the original contribution of this paper. Moreover, *Rybu4WS* Debugger tool was developed to visualize counterexamples or witnesses caught from *DedAn* directly on the original *Rybu4WS* code. The latter feature is unique among model checking tools: they verify the systems but do not allow to interpret the checking results on the source code of the tested system. The projection of the verification result onto the source form of the specification is one of the most significant achievements of the authors.

This paper is organized as follows: Section 2 covers related work of web service composition. Architecture of the Experimentation Framework is in section 3. Section 4 gives a brief description of *IMDS* formalism and *DedAn* tool. Description of *Rybu4WS* and its syntax can be found in section 5. General conversion rules of *Rybu4WS* code to *IMDS* model are described in section 6. Section 7 contains a description of the *Rybu4WS* Debugger tool. Conclusions and possible future development of our experimentation framework are covered in section 8.

II. RELATED WORK

Labeled Transition Systems (LTS) are alternative approaches for modeling Web Service compositions [5] where transitions between states can represent Web Service interactions. In the mentioned paper, model-checking and temporal

logic properties were used to verify the Web Service composition modeled using this approach.

Existing formalisms like CSP [6] or CCS [7] are well designed to model concurrent systems, but they are hardly suitable for distributed systems. They do not possess asynchronous features needed for modeling true distributed systems. Instead, they rely on synchronous communication in the system, which requires that communicating processes reach given states simultaneously before passing a message. Such a scenario is impossible in Web Services or any other distributed system because components are typically placed on separate machines in different locations. They cannot learn about the other party's state in other ways than by message exchange. However, there were attempts of formal software verification based on Service Component Architecture (SCA) [8] converted into CSP specification [9].

Bandera [10] tool allows the creation of a finite-state transition model directly from Java source code that can be verified in the external model checker. The main goal of this project was to provide automated model extraction from software systems that allows easy verification without manual software analysis and model creation. While automated creation of a model from the source code could be very convenient, generated abstraction might affect the model precision. As an alternative to verification, automated WS testing is proposed. Combinatorial method is described in [11] and metamorphic in [12]. Fault injection testing is presented in [13]. Simulation is covered in [14].

There are also languages specifically designed for writing distributed programs, like SR language (Synchronized Resources) [15] that provide various mechanisms used for concurrent process interaction. However, it lacks the ability of formal verification and is not suitable for model checking.

Widely used in industry WSDL [16] format describes Web Service interfaces for other services or applications. Since WSDL is designed to specify the pure interface of Web Services, it is not possible to define the internal behavior of Web Service, which is necessary for verification against deadlocks or checking termination.

A significant number of studies were conducted about Web Service composition, for example hybrid approach [14]. Report [15] presents different automatic composition approaches. TripICS [16] is an example of a real-life application that uses automatic WS composition for planning trips and travels around the world. It is based on the PlanICS framework to solve automatic composition problems, which uses a combination of SMT-solver and genetic algorithms [16]. Automated WS composition for Financial Decision Support is presented in [17]. Semantic modeling is covered in [21][22].

III. ARCHITECTURE OF THE EXPERIMENTATION FRAMEWORK FOR WEB SERVICE COMPOSITION

To provide efficient experimentation with Web Service Composition, a modular but highly integrated system was created. Rybu4WS program is converted to IMDS form and

checked by DedAn verifier, then the witness/counterexample is caught by the Rybu4WS Debugger which can simulate the verification output directly on the source Rybu4WS code.

When DedAn is run with user interface, additional analysis facilities become available, like export to Uppaal for checking huge systems, graphical simulation over system components [18], counterexample animation, and detailed analysis of individual components' behavior.

IV. IMDS AND DEDAN

IMDS [2] formalism is the key element of the experimentation system. Therefore, it will be discussed first. It is a model of a distributed system using that is constructed over a set of actions. The actions are executed in the environment of servers offering services and traveling agents representing distributed computations. The agents use messages for their traveling between servers where partial computations are performed as the execution of actions. A set of messages in given sequential distributed computation forms an agent. The current configuration of a system is defined as a set of states of all servers and a set of current messages of all agents (one message per agent).

Action is a relation between the input pair (message, state) and output pair (new message, new state). The server in a given state accepts the message, which invokes the action. Action execution changes the state of the server and issues a new message. There is a special case on agent termination, which changes only a state without sending a message. The system in IMDS starts from the initial configuration, which consists of initial states for each server and initial messages for each agent.

Formally, the IMDS action is a quadruple of input items and output items $((message, state) \rightarrow (next\ message, next\ state))$ or a triple $((message, state) \rightarrow (next\ state))$ (agent terminates).

The IMDS formalism can well represent Web Service composition due to the following essential features:

- Locality - there is no global or non-local state in the system, all servers are independent.
- Autonomy of decisions - server autonomously determines the order of message acceptance.
- Asynchrony of actions – always a message waits for an appropriate state or a state waits for matching message.
- Asynchrony of communication - messages are sent through a unidirectional asynchronous channel.

In most cases, the states of servers and the messages of agents can be treated as atomic, and actions are defined as a relation in $(M \times S) \times (M \times S)$ which defines input message, input state, output message, and output state of an action. Agent-terminating actions are defined in $(M \times S) \times (S)$. The action extracts the input pair $(message, state)$ from the input configuration and inserts the pair $(new\ message, new\ state)$ into the output configuration. The execution of actions is assumed to be in interleaving semantics [2].

The IMDS models can be verified using the DedAn environment, which allows to find deadlocks or check possible termination in the modeled system. The input of DedAn was designed to structure a set of actions by combining them across servers or agents. It allows defining server and agent types used to instantiate variables of those types along with linking them using formal and actual parameters.

V. RYBU FOR WEB SERVICES (RYBU4WS)

It should be emphasized that the set of actions of Rybu/Rubu4WS specification is exactly the same as the set of actions in IMDS specification after conversion. The main role of higher level language is to ease the programming. The instructions in Rybu/Rybu4WS group sets of actions into more readable high level actions, and chain the actions as in imperative language programming.

A Rybu [4] system consists of two kinds of servers: reactive servers and threads (processing servers from which the agents originate). The agent starts its run in a thread and invokes services offered by the servers by means of messages. Invoked service executes an action on the server, changing its state. A server replies from the executed action by sending a message back to the thread, prolonging its execution. The Rybu4WS was developed for modeling Web Service compositions, overcoming the limitations imposed by Rybu. It features more advanced functionality, such as:

- server-server communication that allows agents to travel between different servers and execute complex scenarios,
- state variables in grouped processes, which enables the communication between different processes without sending actual IMDS messages,
- termination at any point of execution,
- complex code sequences in reactive server actions instead of trivial state mutation and return value.

Like Rybu, the Rybu4WS system consists of reactive servers and processes (in Rybu: thread). The reactive server is a resource that holds a particular state and offers services. Each service can be guarded by a condition over variables and contains a code sequence for further actions. Process consists of a code sequence that the agent executes to invoke services on reactive servers and does not hold any state.

Rybu4WS introduces a third, more advanced feature called group, which is used to group one or more processes. It gives the possibility to declare shared variables, allowing the creation of more sophisticated scenarios where processes use the same variables within a server to cooperate.

The following listing presents example Web Service composition in Rybu4WS. It consists of services necessary to build a simple book shop service – warehouse, payment, bank. Processes are used to represent the user’s behavior.

```

1 type BOOL = { t, f };
2 server Payment {
3   var s: { none, pending, paid };
4   { Init | s == :none } -> { return :ok; }

```

```

5   { Confirm | s == :pending } -> {
6     s = :paid; return :ok;
7   }
8   { IsPaid | s == :paid } -> { return :t; }
9 }
10 server Bank(p: Payment) {
11   var bal: 0..5;
12   var s: BOOL;
13   { Transfer | bal > 0 && s == :f } -> {
14     s = :t; return :confReq;
15   }
16   { Transfer | bal == 0 || s == :t } -> {
17     return :fail;
18   }
19   { Confirm | s == :t && bal > 0 } -> {
20     bal -= 1; s = :f; p.Confirm(); return :ok;
21   }
22 }
23 server Warehouse() {
24   var x: BOOL;
25   { Reserve | x == :f } -> { x = :t; return :ok; }
26   { Reserve|x == :f } -> { return :outOfStock; }
27   { Dispatch | x == :t } -> { x = :f; return :ok; }
28 }
29 server BookShop(w: Warehouse, p: Payment) {
30   { Begin } -> {
31     match w.Reserve() {
32       :outOfStock -> { return :fail; }
33       :ok -> { p.Init(); return :payReq; }
34     }
35   }
36   { End } -> {
37     match p.IsPaid() {
38       :t -> { w.Dispatch(); return :ok; }
39     }
40   }
41 }
42 var p = Payment() { s = :none };
43 var b = Bank(p) { bal = 3, s = :f };
44 var w = Warehouse() { x = :f };
45 var bs = BookShop(w, p);
46 group BookPurchaseScenario {
47   var action: { idle, none, pay } = :idle;
48   process UserWebInterface {
49     match bs.Begin() {
50       :fail -> { action = :none; terminate; }
51       :payReq -> {
52         match b.Transfer() {
53           :confReq -> {
54             action = :pay; bs.End(); terminate;
55           }
56           :fail -> { terminate; }
57         }
58       }
59     }
60   }
61   process UserMobileApp {
62     wait(action != :idle);
63     if (action == :pay) { b.Confirm(); }
64     terminate;
65   }
66 }

```

A. Reactive servers

The reactive server in Rybu4WS consists of variables, actions, and dependencies.

Variables form the internal state of the server, which can be used in conditions for action and can be mutated by the actions code.

An action defines the behavior of a service. It includes an optional predicate, a condition over state variables used to determine whether an agent can execute the given action in the current server state or not, and a code sequence for execution. The code sequence might contain service calls to other servers, variable mutations, return statements, process termination, or conditional statements. Each service has a unique name used by other servers or processes for calling the service. In case a server state satisfies the predicate of more than one action in a given service, the action to execute is chosen non-deterministically. The code sequence is a sequence of statements executed when an action is invoked.

The collection of other servers needed by the given server is called dependencies. Only servers defined in the dependency list can be called from the server.

In order to use reactive servers, an actual server instance must be created. Initial state and required dependencies must be defined for each reactive server instance. This allows creation of many servers with the same behavior but with different initial states or dependencies.

B. Processes

A process is a code sequence with an accompanying agent. It is used as an entry point for agent execution. Each process is converted into one IMDS server and a single IMDS agent. Ungrouped processes (group will be explained later) can only call instantiated reactive servers in the system and cannot define any variables.

C. Groups

A group is a collection of processes and variables. The group's primary goal is to enable processes to use shared variables for sophisticated business scenarios where processes can communicate without sending actual IMDS messages. Agents are instantiated like in ungrouped processes, one agent per one process, meaning that many agents can work simultaneously on the same variables.

VI. CONVERSION TO IMDS

Rybu4WS language is used only for modeling distributed systems and cannot be directly verified against deadlocks or terminations. For the purpose of verification in the DedAn environment, Rybu4WS code must be converted into IMDS equivalent using a set of unambiguous translation rules. More detailed description of Rybu4WS language and architecture of the environment is available in [19].

It is important to note, that during conversion, the original code locations of each statement are preserved in IMDS states and messages. In a later stage, they are used to visually present deadlock or termination/non-termination sce-

narios directly on the Rybu4WS code after verification in DedAn.

Each Rybu4WS reactive server instance is converted to a state machine. Every state machine represents a single IMDS server.

Server variables are converted to IMDS states exactly like in Rybu - they are defined by a Cartesian product of sets of all possible values of the variables in the server.

The process is converted into a state machine and corresponding agent instance. In comparison to Rybu4WS reactive servers, it does not provide any services that could be externally invoked. The process consists of a single code sequence block that is converted into a state machine and provides a special service used by agent as an entry point.

Group is converted into a single, encompassing state machine by merging state machines created for each process. Additionally, the group can contain variables that are converted as in the reactive servers. Many agents can run concurrently in the same group and share variable values without sending any IMDS message. Each action in the encompassing state machine also includes an agent for which this transition is valid, which means that the agent can travel only within his corresponding code sequence.

VII. RYBU4WS DEBUGGER

The Rybu4WS Debugger [20] is a desktop application that allows Rybu4WS code to be loaded and converted it into a corresponding IMDS representation for the purpose of verification in the DedAn environment. DedAn is invoked automatically (or manually if advanced analysis options are needed) and finds deadlocks/checks termination automatically. It is worth emphasizing that partial deadlocks are identified as well. During verification, a counterexample/witness is elaborated, and visualized in a user-friendly way for the manual analysis. This reverse mapping of the sequence of action onto the source code is achieved because actions in Rybu4WS program are expressed in a more abstract and easily readable form than in IMDS specification. However, the sets of actions in Rybu4WS and IMDS are exactly the same, and the semantics of both specifications is equal.

VIII. CONCLUSIONS

The growing number of designed Web Services requires more and more assistance in the programmer's activities, including verification of whether the designed services behave safely (free from deadlocks), whether they finish in inevitable success (process termination), or whether there is even a possibility of success. Tools based on temporal logic are used to verify such behavior.

The formalism used to describe services should be well-suited to distributed systems: support asynchrony, locality of actions, and autonomy of nodes. Ideally, the specification language can be explicitly used for formal verification. Such a modeling method is IMDS, for which the DedAn verification environment has been built. However, the DedAn input language does not fully meet the requirements of program-

mers; therefore the Rybu language was created and its successor – Rybu4WS. Conveniently for the programmer, it combines the basic IMDS paradigm, adding the possibility of coupling actions in reactive servers. It was achieved using a syntax close to the programmers’ habits, using typical control statements such as conditional branching, loops, and response handlers. Shared variables in process groups allow to easily communicate by mutation of variable values.

The severe limitation of Rybu – allowing the server to be called only from a process – was solved: now a call chain can be created. Additionally, it is possible to terminate the process without returning it to the home server. Communication between servers, declaring shared variables for processes, and termination at any point of execution gives the ability to model Web Service compositions.

The Rybu4WS Debugger tool provides a user-friendly interface that allows analyzing counterexamples similarly to debuggers in usual programming environments. It is a backward engineering principle, seldom observed in typical verifiers: they produce counterexamples or witnesses that are not easy to analyze in the context of the source code.

Rybu4WS can be used to model a wider variety of concurrent distributed systems than just Web Service themes. The development needs of such systems would require additional programming elements, which could eventually lead to the creation of a Domain Specific Language (DSL) family that might be the subject of further research. It is currently impossible to create a circular dependency between reactive servers, meaning that callbacks or recursive calls are not supported. Also, it is not possible for a single agent to ”split” and perform multicast action, i.e., calling multiple services in a parallel manner. That would allow the creation of agents traveling in the distributed system and not returning to the place they originate from.

REFERENCES

- [1] B. AL-Shargabi, A. El Sheikh, and A. Sabri, “Web Service Composition Survey: State of the Art Review,” *Recent Patents Comput. Sci.*, vol. 3, no. 2, pp. 91–107, Jun. 2010. doi:10.2174/2213275911003020091
- [2] W. B. Daszczuk, “Specification and Verification in Integrated Model of Distributed Systems (IMDS),” *MDPI Comput.*, vol. 7, no. 4, pp. 1–26, Dec. 2018. doi:10.3390/computers7040065
- [3] W. B. Daszczuk, “Using the Dedan Program,” in *Integrated Model of Distributed Systems*, Cham, Switzerland: Springer Nature, 2020, pp. 87–97. doi: 10.1007/978-3-030-12835-7_6
- [4] W. B. Daszczuk, M. Bielecki, and J. Michalski, “Rybu: Imperative-style Preprocessor for Verification of Distributed Systems in the Dedan Environment,” in *KKIO’17 – Software Engineering Conference, Rzeszów, Poland, 14-16 Sept. 2017*, 2017, pp. 135–150. <https://arxiv.org/abs/1710.02722>
- [5] M. Ghannoudi and W. Chainbi, “Formal verification for Web service composition: A model-checking approach,” in *2015 International Symposium on Networks, Computers and Communications (ISNCC), Yasmine Hammamet, Tunisia, 13-15 May 2015*, 2015, pp. 1–6. doi: 10.1109/ISNCC.2015.7238576
- [6] C. A. R. Hoare, “Communicating sequential processes,” *Commun. ACM*, vol. 21, no. 8, pp. 666–677, Aug. 1978. doi:10.1145/359576.359585
- [7] R. Milner, *A Calculus of Communicating Systems, LNCS vol. 92*, vol. 92. Berlin, Heidelberg: Springer Berlin Heidelberg, 1980. ISBN 978-3-540-10235-9
- [8] H. Paik, A. L. Lemos, M. C. Barukh, B. Benatallah, and A. Natarajan, “Service Component Architecture (SCA),” in *Web Service Implementation and Composition Techniques*, Cham: Springer International Publishing, 2017, pp. 203–250. doi: 10.1007/978-3-319-55542-3_8
- [9] W. Chargui, T. S. Rouis, M. Kmimech, M. T. Bhiri, L. Sliman, and B. Raddaoui, “Towards a formal verification approach for service component architecture,” in *SOMET 2017: 16th International Conference on Intelligent Software Methodologies, Tools, and Techniques, Kitakyushu, Japan, 26-28 Sept 2017*, 2017, pp. 466–479. doi: 10.3233/978-1-61499-800-6-466
- [10] J. C. Corbett, M. B. Dwyer, and J. Hatcliff, “Bandera: a source-level interface for model checking Java programs,” in *22nd international conference on Software engineering - ICSE ’00, Limerick, Ireland, 4-11 June 2000*, 2000, pp. 762–765. doi: 10.1145/337180.337625
- [11] I. Bluemke, M. Kurek, and M. Purwin, “Tool for Automatic Testing of Web Services,” in *5th International Workshop Automating Test Case Design, Selection and Evaluation, FEDCSIS, Warsaw, Poland, 7-10 Sept 2014*, 2014, pp. 1553–1558. doi: 10.15439/2014F93
- [12] I. Bluemke and A. Sawicki, “Tool for Mutation Testing of Web Services,” in *13th DEPCOS/Reclomex, Brunów, Poland, 2-6 July 2018*, 2019, pp. 46–55. doi: 10.1007/978-3-319-91446-6_5
- [13] S. Ilieva, I. Manova, and D. Petrova-Antonova, “Towards a methodology for testing of business processes,” in *7th Federated Conference on Computer Science and Information Systems, FEDCSIS, Wroclaw, Poland, 09-12 Sept 2012*, 2012, pp. 1315–1322. <https://ieeexplore.ieee.org/document/6354354>
- [14] T. Preisler, T. Dethlefs, and W. Renz, “Simulation as a Service: A Design Approach for large-scale Energy Network Simulations,” in *10th Federated Conference on Computer Science and Information Systems, FedCSIS, Lodz, Poland, 13-16 Sept 2015*, 2015, pp. 1765–1772. doi: 10.15439/2015F116
- [15] G. R. Andrews et al., “An overview of the SR language and implementation,” *ACM Trans. Program. Lang. Syst.*, vol. 10, no. 1, pp. 51–86, Jan. 1988. doi:10.1145/42192.42324
- [16] E. Christensen, F. Curbera, G. Meredith, and S. Weerawarana, “Web Services Description Language (WSDL),” 2001. <https://www.w3.org/TR/wsdl.html>
- [17] L. Belava, “Concept of Platform for Hybrid Composition, Grounding and Execution of Web Services,” in *11th Conference on Advanced Information Technologies for Management, FEDCSIS, Kraków, Poland, 8-11 Sept 2013*, 2013, pp. 1071 – 1077. https://annals-csis.org/Volume_1/pliks/190.pdf
- [18] G. Baryannis and D. Plexousakis, “Automated Web Service Composition: State of the Art and Research Challenges,” 2010. https://publications.ics.forth.gr/tech-reports/2010/2010.TR409_Automated_Web_Service_Composition.pdf
- [19] A. Niewiadomski, P. Switalski, M. Kowalczyk, and W. Penczek, “TripICS - a Web Service Composition System for Planning Trips and Travels,” *Fundam. Informaticae*, vol. 157, no. 4, pp. 403–425, Jan. 2018. doi:10.3233/FI-2018-1635
- [20] I. Pawełszek, “Integrating Semantic Web Services into Financial Decision Support Process,” in *11th Conference on Advanced Information Technologies for Management, FEDCSIS, Gdansk, Poland, 11-14 Sept 2016*, 2016, pp. 1189–1198. doi: 10.15439/2016F99
- [21] S. De, P. Barnaghi, M. Bauer, and S. Meissner, “Service modelling for the Internet of Things,” in *6th Federated Conference on Computer Science and Information Systems, FedCSIS, Szczecin, Poland, 18-21 Sept 2011*, 2011, pp. 949–955. <https://ieeexplore.ieee.org/document/6078180>
- [22] S. Demirkol, M. Challenger, S. Getir, T. Kosar, G. Kardas, and M. Mernik, “SEA_L: A Domain-specific Language for Semantic Web enabled Multi-agent Systems,” in *7th Federated Conference on Computer Science and Information Systems, FEDCSIS, Wroclaw, Poland, 09-12 Sept 2012*, 2012, pp. 1373–1380. <https://ieeexplore.ieee.org/document/6354358>
- [23] W. B. Daszczuk, “Graphic modeling in Distributed Autonomous and Asynchronous Automata (DA³),” *Softw. Syst. Model.*, vol. 20, no. 5, pp. 363–398, 2021. doi:10.1007/s10270-021-00917-7
- [24] S. Katra, “Specification and verification of Web Service composition in DedAn environment,” MSc thesis, Dept. of Electronics and Information Technology, Warsaw University of Technology, 2022.

Beyond Low-Code Development: Marrying Requirements Models and Knowledge Representations

Kamil Rybiński, Michał Śmiałek

Warsaw University of Technology, Poland

Email: {kamil.rybinski, michal.smialek}@pw.edu.pl

Abstract—Typical Low-Code Development platforms enable model-driven generation of web applications from high-level visual notations. They normally express the UI and the application logic, which allows generating the frontend and basic CRUD operations. However, more complex domain logic (data processing) operations still necessitate the use of traditional programming. This paper presents a visual language, called RSL-DL, to represent domain knowledge with complex domain rules aligned with requirements models. The language synthesises and extends approaches found in knowledge representation (ontologies) and software modelling language engineering. Its purpose is to enable a fully automatic generation of domain logic code by reasoning over and reusing domain knowledge. The language’s abstract syntax is defined using a meta-model expressed in MOF. Its semantics is expressed with several translational rules that map RSL-DL models onto typical programming language constructs. The rules are explained informally in natural language and formalised using a graphical transformation notation. It is also supported by introducing an inference engine that enables processing queries to domain models and selecting appropriate invocations to generated code. The presented language was implemented by building a dedicated model editor and transformation engine. It was also initially validated through usability studies. Based on these results, we conclude that declarative knowledge representations can be successfully used to produce imperative back-end code with non-trivial logic.

I. INTRODUCTION

THE TERM “Low-Code Software Development” (LCSD) has emerged as a new approach to application development where only a limited amount of coding is required. Since its emergence around seven years ago [25], the term was used in the industry to label cloud-based development platforms that use visual notations to reduce the need for traditional programming. Research on low-code approaches is currently yet sparse. However, just recently, it has been observed that LCSD can be seen as a subdomain [7] or as overlapping [27] with the Model-Driven Software Development (MDS, MDD), where it concentrates on automatic generation of data-rich web/mobile applications from visual specifications (models).

Our research was done in the context of the ReDSeeDS¹ platform [33]. The system was created before the emergence of the low-code movement, but it certainly fulfils the definition of LCSD. It uses precisely specified requirements models: use cases, scenarios in constrained language, and visual domain vocabularies. Similarly to low-code platforms, these

¹<https://github.com/smialekm/redseeds>

artefacts are represented with a visual language called RSL (Requirements Specification Language) [21]. Specifications expressed in RSL allow for the generation of fully functional UI and application logic code for data-rich web applications. However, based on RSL alone, one cannot generate data processing (domain logic) code beyond simple CRUD, and data persistence operations [39]. Thus, similarly to other low-code platforms, more complex data processing has to be coded manually.

In this paper, we raise the question of representing more complex domain logic at the level of abstraction used by low-code and requirements-based MDD approaches. Inspiration for responding to this question can be drawn from research on ontologies and knowledge representations. These approaches enable the representation of domain knowledge, independently of any technology and any particular problem domain. Our research aims at investigating how the features of ontology-based domain logic representations, especially their reasoning capabilities, can be applied in the context of LCSD.

We present an extension to the RSL mentioned above, which we call RSL-DL (RSL Domain Logic, see the initial version of the language in our previous work [28]). The new language draws several of its constructs from ontology-based knowledge representation approaches. It then extends and combines them with MDD technologies to provide full code generation capabilities. What is important, RSL-DL allows for the generation of fully operational code directly from general domain rules (descriptions of reality) as required by specific requirements models (use cases and their logic). Hence, such generated back-end code is fully compatible with the front-end code generated from RSL specifications. Through this, we demonstrate a visual extension to a low-code language (RSL) that has the potential of eliminating the need to code (in a traditional sense) complex data processing services.

II. MOTIVATION AND RELATED WORK

The term “low-code” was probably first used in a Forrester report [25] only in 2014. Current studies report numerous industry-grade low-code platforms used extensively by professionals [29]. Sparse research results on LCSD can be supported by previous and current research on Model-Driven Web Engineering [17] which can be seen as a predecessor and now a synonym for LCSD. A study by Wakil and Jawani [38]

shows that research on MDWE is already quite broad and mature. LCSD and MDWE approaches are typically based on some form of visual notation (language). Such notations offer high-level representations of the flows of interaction between application users and the system under development. A prominent example in the MDWE domain is the Interaction Flow Modeling Language (IFML) [4]. Other examples – in the LCSD domain – are the Business Process Technology (BPT) language [13] and the Mendix notation [12].

LCSD/MDWE systems have limited capabilities regarding the generation of the domain/business logic code, or more broadly – the system’s back-end. Current LCSD/MDWE languages can support generation of code for elementary CRUD (Create-Read-Update-Delete) operations [3], [26]. Generation of code for more complex data processing (general Domain Logic - DL) is limited by the information scope of the visual language constructs. In this research, we propose new constructs that significantly extend capabilities to generate complex domain logic code. What is important, these new constructs are domain-agnostic, as contrasted with various domain-specific and often very formalised notations (see, e.g. work by Hinchey et al. [14] and Brito et al. [5]).

Our research is in line with the work by Atkinson et al. [2] that shows significant similarities between ontologies and models. The authors argue that the concept of ontology constitutes a subset of the concept of model. Also, Henderson-Sellers [11] points out that a combination of models and meta-models with domain ontologies is helpful in representing vocabularies for specific problem domains. He argues that modelling languages such as UML can describe domain knowledge, but they need particular extensions to provide adequate reasoning support. Our approach goes in this specific direction, as it extends RSL, which is also an extension to UML. In summary, the above discussions give good motivation for our work, where a modelling language that combines ontology constructs is applied to generate code directly from requirements.

Marrying ontologies with models allows applying model-driven techniques and especially model transformations. Appropriate works include more general discussions on introducing comprehensive formalised propositions on meta-models for ontology languages [23], [9]. An example of such a language is CoCoViLa by Haav, and Ojamma [10]. Our current work can be compared or even contrasted with such approaches, as it introduces a common meta-model for a semantically rich language that can express any problem domain. In this context, an essential feature of our approach is its extensive reliance on inference mechanisms, especially for generating data processing code. It is somewhat similar to business rule engines [8]. However, instead of interpreting them during runtime, it generates code fully integrated with the rest of the system. Similarly, our solution can be compared with the approaches that enable code generation directly from ontologies. Stevenson and Dibson [34] propose a tooling framework for generating Java code from OWL specifications. Another example is work by Völkel and Sure

[36] in which Java-based APIs are generated directly from ontologies expressed in RDF Schema.

III. RSL: LOW-CODE AT THE REQUIREMENTS LEVEL

The necessary background to our research on RSL-DL is the Requirements Specification Language and its tooling environment (ReDSeeDS). As we strive to achieve compatibility between these two languages, some aspects of RSL-DL were strongly influenced by RSL. The most important for this purpose are use case scenarios, like the example one shown in Fig. 1. Scenarios consist of sequences of simple subject-verb-object sentences of various types. The most important of them from the point of view of this paper is the ‘Query’ type sentences. These sentences define actions of the system that are performed on certain data elements.

According to RSL semantics rules [31], we can transform scenarios into code. Fig. 2 presents fragments of an application logic class generated from the presented scenario. The class contains methods for handling user interactions, as specified by the ‘Select’ sentences in the scenarios. For instance, the sentence no. 1 is translated into the “summarizeSemesterTriggered” method. Contents of these methods reflect consecutive sentences in the scenarios. ‘Query’ and CRUD type sentences are translated into calls to back-end service operation. For

Name:	Action Type
Summarize semester	
precondition:	
1. Dean's office employee selects summarize semester	Select
2. System fetches students list	Read
3. System prepares semester summarization data	Query
4. System shows semester summarization window	Show
5. Dean's office employee selects accept summarization	Select
6. System closes semester summarization window	Close
7. System realizes semester summarization data	Update
8. System shows semester summarized message	Show
final: success	

Fig. 1. Example scenario

```
public class SummarizeSemesterPresenter extends
    AbstractUseCasePresenter {
    // ...
    public void summarizeSemesterTriggered(){
        studentListDTO =
            service.readStudentList();
        semesterSummarizationDataDTO =
            service.preparesSemesterSummarizationData
                (pstudentListDTO);
        view.showShowSemesterSummarizationWindow(this);
        pageOpened();
    }

    public void saveFinalGradeTriggered(){
        view.closeSemesterSummarizationWindow();
        pageClosed();
        service.realizesSemesterSummarizationData
            (semesterSummarizationDataDTO);
        view.showSemesterSummarizedMessage();
    }
    // ...
}
```

Fig. 2. Presenter code generated for the use case scenario

```

public class ServiceImpl implements IService {
    // ...
    List<SemesterSummarizationDataItemDTO>
    preparesSemesterSummarizationData
        (List<StudentListItemDTO> studentListDTO)
    {}

    void realizesSemesterSummarizationData
        (List<SemesterSummarizationDataItemDTO>
         semesterSummarizationDataDTO)
    {}
    // ...
}

```

Fig. 3. Backend access code generated for the example scenario

instance, the sentence no. 7 is transformed into a call to the “realizesSemesterSummarizationData” service. UI presentation sentences are translated into calls to the View layer. A more detailed discussion of the rules and generated code, including code of the View layer, is presented elsewhere [32].

The relevant parts of the back-end service code generated from the RSL scenario is presented in Fig. 3. It contains an interface implementation with empty methods. The operation parameters are determined from scenario sentences before the appropriate calls. In further sections, we will present the syntax and semantics of RSL-DL that will fill the currently empty method bodies.

IV. RSL-DL SYNTAX

The syntax of RSL-DL aims to represent all information important from the point of view of code generation. It includes detailed dependencies between individual domain elements and proper definitions of the elements themselves. Fig. 4 presents an elementary example of concrete syntactic elements of the language. It contains definitions of four “Identity” type entity notions (student, course, partial grade, weighted grade), describing concrete objects in the specific problem domain. Notions can also have conditions, and in our example, we can see one kind of condition: “inheritance”. Thus, the condition for the “partial grade” notion is that it must follow all the rules for the “weighted grade” notion. In addition to entity notions, we can define property notions, like “grade weight” in Fig. 4. This kind of notions define concrete atomic values and can be used as attributes of other notions, which can be indicated by ‘attribute links’ (lines with a diamond shape).

Relationships in RSL-DL (see “grading” in Fig. 4) are represented by hexagons and can link many notions. To some

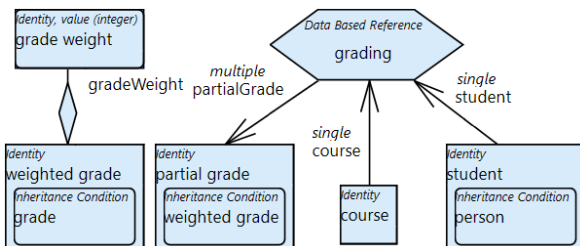


Fig. 4. RSL-DL concrete syntax example

extent, this syntax resembles that of UML’s n-ary associations. In our example, the relationship is of type “Data-Based Reference” which is a basic type that reflects the situation where references between objects are contained in their data (e.g. in their attributes). In our current example, “student” and “course” contribute to the relationship “grading” that results with a “partial grade”. Arrow directions distinguish between types of notion participations in the specific relationship. Since a given student can have many partial grades in a course, then the particular participation is marked as ‘multiple’.

The abstract syntax for the above-presented core language elements is presented in Fig. 5a using the MOF notation [22]. The meta-class “DLNotion” represents notions and the meta-class “DLRelationship” represents dependencies between them. Concrete participations of notions in relationships are represented by the meta-class “DLRelationshipParticipation”. The meta-model contains two types of such participations – standard and auxiliary. Standard ones correspond to the main subjects of relationships and are denoted with solid arrows in concrete syntax. Auxiliary ones point to elements that define relationship contexts and are denoted with dashed lines. For example, one could use a relationship context to indicate which object should be used when computing values based on that object’s attributes. In this case, the attributes participate through standard participations, and their ‘parent’ participates through auxiliary participation.

Besides notions, there is a special kind of relationship participants – primitives (“DLPrimitive” meta-class). These elements define general concepts that do not have concrete instances. Examples of such primitives in RSL-DL are “current date”, “number Pi” and “Planck constant”.

As indicated above, notions can have types. The first one (“identity”) was explained in the example above. The “template” type indicates templates that can be used to simplify defining other notions. These two types correspond approximately to concrete and abstract classes of e.g. UML. Two other types define notions whose representatives’ (objects’) roles can change during their lifetime. It is inspired by ontology-based inference engines with their capabilities to “discover” object types or change them dynamically. The “inferred role” type indicates roles that can be inferred, e.g. from various status attributes of an object. The “assigned role” type indicates roles that can be explicitly changed during the lifetime of an object.

More details related to the syntax for notions are shown in Fig. 5b. The “DLProperty” meta-class is used to denote notions with concrete atomic values or value sets. The “DLEntity” meta-class denotes more complex notions that cannot be reduced to single values. The “DLAttributeLink” meta-class allows indicating attribute dependencies between notions. Such links can be marked as ‘derived’, which means that their values need to be inferred from other notions. An important type of notion features are conditions (“DLCondition” meta-class). Their role is to further detail notion characteristics. Apart from the previously described ‘inheritance condition’, two additional condition types exist. The ‘identity condition’ type defines conditions that have to be fulfilled for a given notion’s

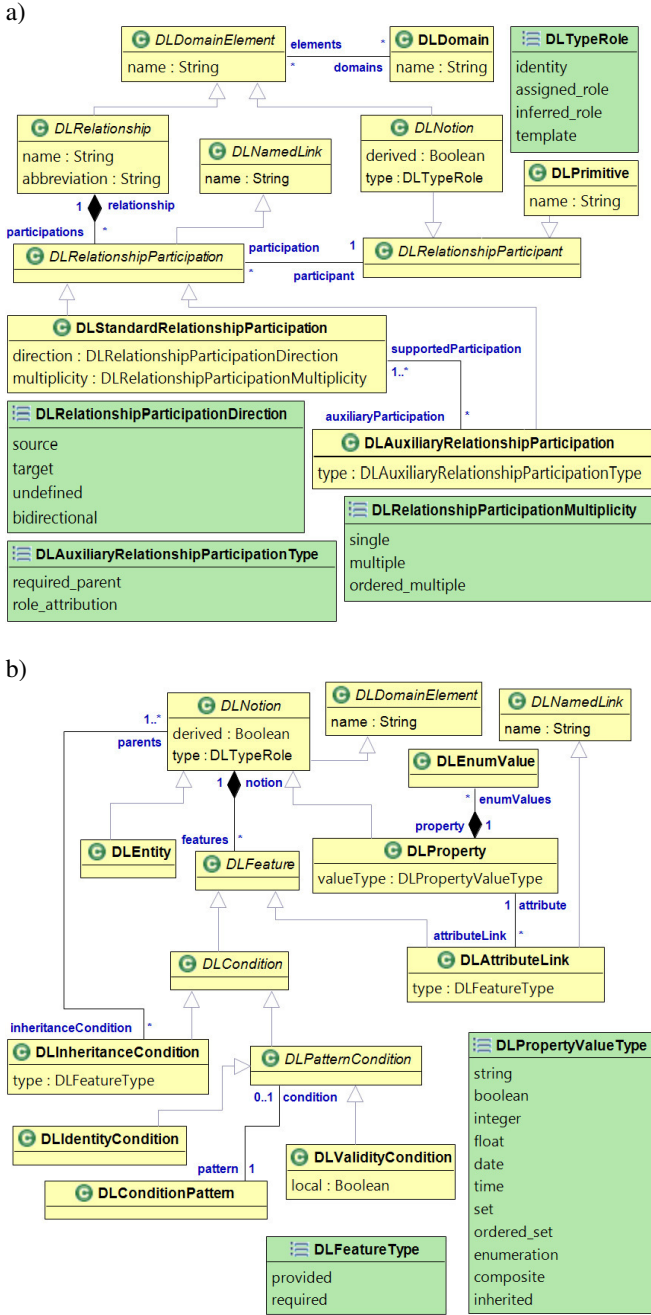


Fig. 5. Meta-model fragments for core RSL-DL elements (a), and for notions (b)

object to make sense. The ‘validity conditions’ type defines conditions that denote the correctness of a given notion’s object. In general, there can exist objects that meet appropriate identity conditions but do not meet validity conditions and thus are treated as invalid but belonging to the given notion. We should note that conditions do not include graphical links to other model elements. It is due to their potential complexity and interweaving. Thus, for instance, inheritance was not represented using a simple arrow as in UML.

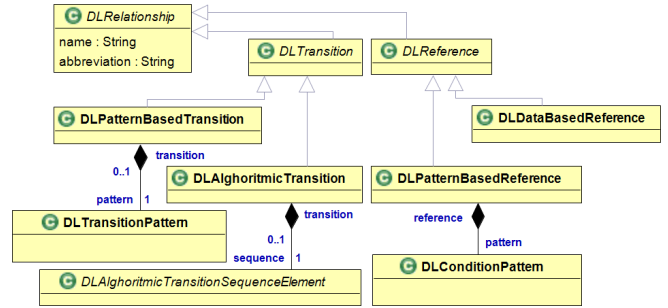


Fig. 6. Meta-model details for relationships

Fig. 6 presents the hierarchy of relationships found in RSL-DL. From the conceptual point of view, two main types of dependencies exist between notions in the problem domain that are significant for code generation. It is reflected in RSL-DL through dividing relationships into two categories: transitions (“DLTransition”) and references (“DLReference”). Transitions describe how to obtain notion objects based on other notion objects. References describe specific roles played by objects in relation to other objects. Both relationship categories are further divided based on how they are defined. ‘Transitions’ can be described using simple rules (“DLPatternBasedTransition”) or algorithms consisting of many steps (“DLAlgorithmicTransition”). ‘References’ can be described using rules that define certain conditions (“DLPatternBasedReference”) or take the form of the previously described data-based references (“DLDataBasedReference”). This division of references is inspired by the division into fact and rule spaces found in ontologies. In practice, of all these relationships, the algorithmic transitions are not preferred as they are not fully declarative and thus arguably less usable [15], [37].

Figs. 5b and 6 contain two additional elements: “DLTransitionPattern” and “DLConditionPattern”. Each instance of these two meta-classes contains a string with a textual condition, a specific condition type and optionally – the condition’s subject link. The syntax of the condition is expressed in a language based on the notation used in the Symja library [18]. Some examples of several types of these patterns are given in section VI in descriptions of the generated code.

V. TRANSLATIONAL SEMANTICS RULES

Of the many approaches to define semantics for RSL-DL (treated as a programming language [30]) we choose the translational method, which is more in line with the model-driven paradigm. This approach defines rules that translate specific patterns of RSL-DL constructs into fragments of Java code. Each rule has an informal textual description and is formalised as a procedure in the MOLA graphical model transformation language [16].

Full specification of semantics for RSL-DL consists of 16 translational rules (all the details can be accessed in the Supplement²). The first ten rules define the generation

²<https://github.com/smialekm/redseeds/tree/main/RSL-DL>

of the target Java class structure, including their fields and method signatures. These rules depend only on the structure of notions and relationships between them, found in a particular RSL-DL model. The following two rules additionally use an inference engine and are used to generate method bodies. Rules 13-16 further add to the generation of method bodies. They use a symbolic computation library to transform Symja-based formulas found in pattern condition expressions into the contents of method bodies. The results are used directly or as part of a loop or a condition depending on the pattern type. In summary, each ‘non-trivial’ notion in the source RSL-DL model produces two Java classes. One class represents (in simplified terms) a data transfer object (DTO) corresponding to the given notion. The other class is a utility class that holds various data handling methods. These classes are appropriately amended with CRUD and condition-related operations. The ‘‘DTO’’ classes are also organised in an appropriate inheritance hierarchy. Additionally, supportive classes are created for all the relationships in the model. These classes contain methods that return objects participating in relevant relationships.

As introduced above, all the rules are formalised using MOLA procedures. MOLA uses a declarative-imperative visual syntax presented in Fig. 8 and 9. Its imperative flow definition is based on a notation resembling activity diagrams in UML. Arrows denote control flow. Iteration ‘actions’ are denoted with thick black frames. Rule ‘actions’ constitute the declarative part of the language. Each rule contains a query on objects expressed through a diagram resembling a UML object diagram combined with a MOF meta-model diagram. Black solid lines denote queried objects, while red dashed lines denote created objects. More details and a tutorial can be found in the MOLA handbook [1].

For brevity, we will limit our presentation of rule formalisation to only rule 11. To implement it, we need to use a dedicated inference engine, implemented as part of this work. The engine processes queries derived from ‘Query’ type scenario sentences (see again Fig. 1). For each such query, it produces a sequence of inference rules, where each of the rules is based on domain elements defined within an RSL-DL model. The appropriate sequences can be represented with a meta-model shown in a simplified form in Fig. 7. The meta-model uses a structure of nested ‘‘Rule’’ meta-classes to reflect appropriate sequences of inference invocations needed to solve specific problems. Each ‘‘Rule’’ points to a domain

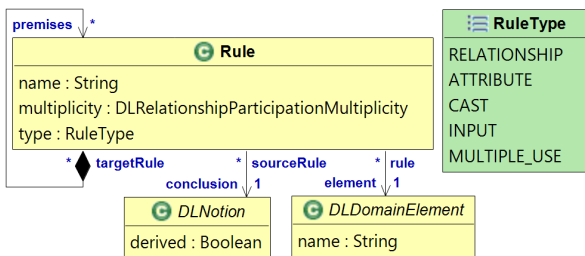


Fig. 7. Inference rule meta-model

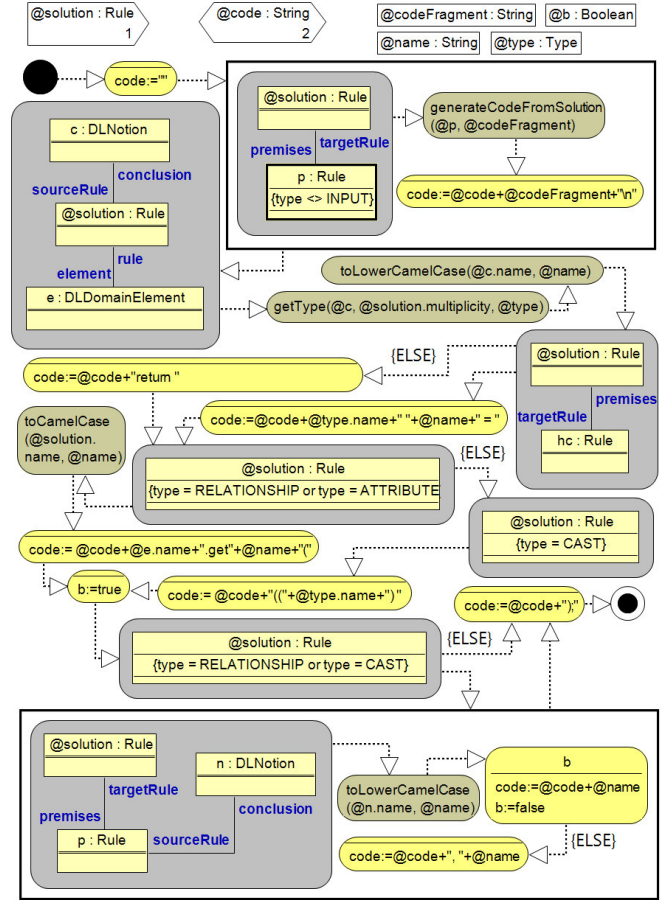


Fig. 8. Algorithm for generating the method body in Rule 11

element (‘‘element’’) that is the basis for generating a specific method according to one of the previous rules. The ‘‘type : RuleType’’ meta-attribute defines the concrete type of such generation. This ‘‘Rule’’ meta-class also points to a ‘conclusion’ that constitutes a specific notion reflecting objects being the inference results. Furthermore, the given rule’s ‘premises’ constitute other rules, preceding this rule in the rule sequence. Base premises that reflect query parameters are represented as additional ‘artificial’ rules.

The algorithm that generates the respective sequence of method invocations from the inference rule structure is presented in Fig. 8. It starts from invoking itself (‘generateCodeFromSolution’) recursively for all the ‘premises’ of the current rule and joining code generated from these premises. If the current element is used as a ‘premise’ in other rules, a proper variable is declared based on the object corresponding to the rule’s ‘conclusion’. Otherwise, a ‘return’ statement is generated. In both cases, the way to obtain the assigned or the returned value depends on the type of the rule. In most cases, such a value is obtained by invoking an appropriate ‘get’ method. This method retrieves an object that corresponds to the ‘conclusion’ and is contained in the class derived from the rule’s ‘element’. Besides, the ‘get’ method’s call accepts

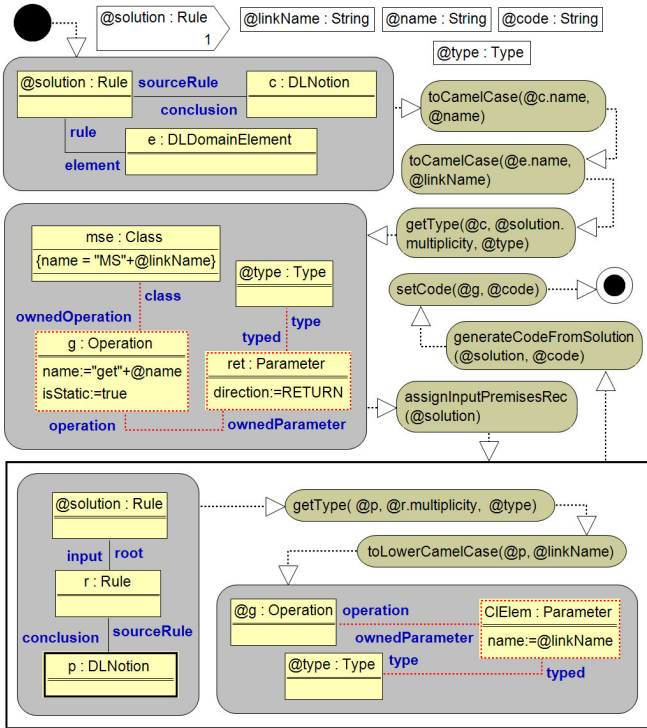


Fig. 9. Formalization of transformation rule 11

parameters that correspond to the rule’s premises.

The actual formalisation of Rule 11 that uses the above algorithm is presented in Fig. 9. It defines a procedure for creating a method, where the method’s contents are generated with the algorithm. The procedure starts by creating a method with the name based on the ‘conclusion’ of the final rule and having the prefix “get”. This method is placed in the class derived from the domain element pointed to by this final rule. The return type of this method again corresponds to the rule’s ‘conclusion’, and the method’s parameters are based on the ‘premises’ within the whole rule sequence.

VI. CASE STUDY

This section will present a selected fragment of a more extensive case study that illustrates several important uses of RSL-DL. The case study refers to the functional requirements specification presented in Section III. Here we will concentrate only on presenting examples of the various concrete RSL-DL language constructs, generated domain logic code, and references to application logic code from Section III.

Fig. 10 involves constructs for checking specific conditions. The model contains elements that describe information related to checking whether the given student is eligible to get a registration for the next semester. It consists of three basic notions – ‘student’, ‘course’ and ‘final grade’. All of them are connected by the “final grading” relationship, which is a data-based reference. It indicates that information about concrete dependencies between representatives of these notions is stored in some data objects. Finally, we define the ‘student to

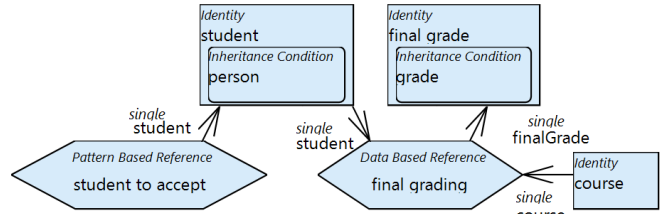


Fig. 10. RSL-DL model defining eligibility of students to be registered for the next semester

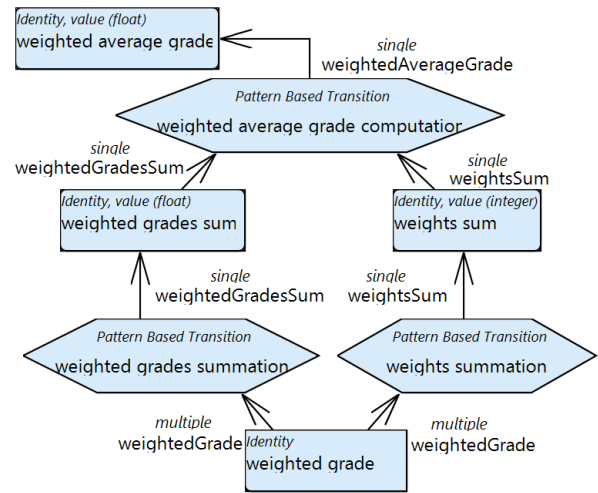


Fig. 11. RSL-DL model containing knowledge about computing of weighted average grades

accept’ relationship that is used to define conditions about students’ eligibility to be registered for the next semester. The concrete condition embedded in this relationship (not shown here) requires that all the final grades for the student’s courses have a value of at least 3 (minimum passing level). The relationship has only one participant – the student, that participates as its target.

Fig. 11 involves constructs for computing values. It contains the ‘weighted average grade computation’ relationship that contains a transition pattern with an equation that computes the weighted average grade. This equation requires two other values represented by the notions ‘weighted grades sum’ and ‘grades sum’. Therefore, the model contains also transitions that allow for the computation of these two values.

The final part of the presented model fragment involves constructs for modifying complex objects and is shown in Fig. 12. The whole modification is handled by the ‘summarize student after semester’ transition, which requires two other transitions: “accept student”, and “fail student”. These transitions will be used interchangeably, depending on the fulfilment of a condition. This condition refers to the ‘student to accept’ relationship presented in Fig. 10. If the student meets the ‘student to accept’ relationship, the ‘accept student’ transition is invoked, while in the opposite case, the ‘fail

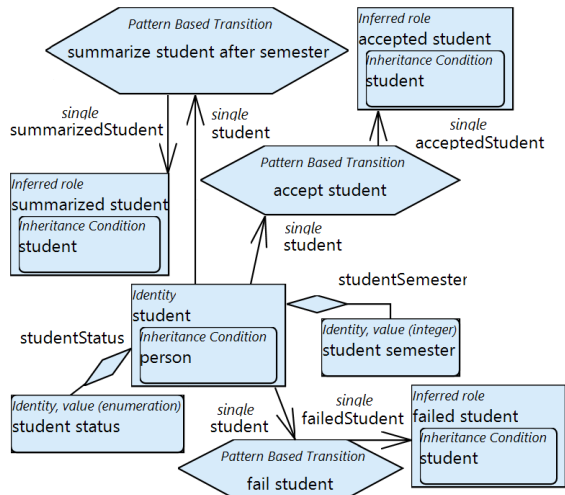


Fig. 12. RSL-DL model containing knowledge about the rules for promoting students to the next level of studies

```

public class MSStudentToAccept {
    public static boolean checkStudentToAccept(
        IMStudent student){
        for (IMFinalGrade $_iter:
            student.getFinalGrades())
            if (!$_iter.getGradeValue()>=3)
                return false;
        return true;
    }
    public static List<IMStudent> getStudents(){
        List<IMStudent> students
            = MSStudent.getStudents();
        List<IMStudent> result
            = new ArrayList<IMStudent>();
        for (IMCourse $_iter:students)
            if (checkStudentToAccept($_iter))
                result.add($_iter);
        return result;
    }
}

```

Fig. 13. Code for checking students' registration eligibility

student' transition is invoked.

The next step in our case study is to generate code. The above specification in RSL-DL was formulated within a dedicated RSL-DL tool. The tool also includes a code generation engine that implements all the rules introduced in the previous section. Here we will present some key fragments of code generated from the above-presented excerpts of the dean office model.

Fig. 13 shows code generated on the basis of the model from Fig. 10. The actual class (MSStudentToAccept) is generated per rule no. 9 (generate classes and static methods from relationships). It is a supportive class that corresponds to the 'student to accept' relationship and contains only static methods. The 'checkStudentToAccept' method was generated based on rule no. 10 (generate existence checking methods). It checks for the eligibility of a given Student to be accepted for the next semester. The student is passed as the parameter of this method. It can be noted that the type of this parameter ('IMStudent') is the class corresponding to the 'student'

```

public class MSWeightedAverageGradeComputation {
    public static double
        getWeightedAverageGrade(double weightedGradesSum,
            int weightsSum){
        return weightedGradesSum/weightsSum;
    }
    public static double
        getWeightedAverageGrade(List<IMPartialGrade> partialGrades){
        List<IMWeightedGrade> weightedGrades
            = new List<IMWeightedGrade>(partialGrades);
        double weightedGradesSum
            = MSWeightedGradesSummation
                .getWeightedGradesSum(weightedGrades);
        int weightsSum = MSWeightsSummation
            .getWeightsSum(weightedGrades);
        return getWeightedAverageGrade(weightedGradesSum,
            weightsSum);
    }
}

```

Fig. 14. Code for computing weighted average grades

notion, generated according to rule no. 1 (generate classes from notions).

The method returns a logical value reflecting the result of the eligibility check. The actual check is based on the contents of the condition pattern in the 'student to accept' relationship (not shown in Fig. 10). This pattern is defined through three values: 1) formula 'gradeValue(\$)>=3', 2) type 'universal quantification', and 3) subject link 'finalGrading(student)' that relates to the 'final grading' relationship. Note that the '\$' sign denotes the target of the 'final grading' relationship, which is the 'final grade' notion. The above formula was transformed into the appropriate 'if' statement in Fig. 13. The 'for' loop is generated based on rule no. 13 (generate condition checking code from condition patterns), considering the above pattern type. This way, the eligibility check is done for all the grades of a particular student.

The second method of this class ('getStudents') applies the above eligibility check to all the students. This method was generated based on rule no. 14 (generate object filtering code from condition patterns), and it filters out all the subjects (here: students) that fulfil an appropriate condition pattern (here: student eligibility check). We should also note that the code for obtaining the list of all students was generated using rule no. 16 (generate auxiliary code).

Fig. 14 shows code generated on the basis of the model from Fig. 11, using data structured according to Fig. 4. This time, the situation is somewhat different to that in the previous code fragments. This part of the code results from answering a query that asks to compute the 'weighted average grade' for the given set of grades. It contains two overloaded methods ('getWeightedAverageGrade'). The second one (with the 'List' parameter) is generated according to rule no. 11. It accepts a list of partial grades and produces a specific average value. The method contains a sequence of method calls that reflect the sequence of inference rules returned by the inference engine (see Fig. 8). The first one of the overloaded methods is called from the second one. Its signature was generated according to rule no. 9. Its body was based on translating a transition pattern with the 'simple' formula 'weightedGradesSum/weightsSum'

```

public class MSummarizeStudentAfterSemester {
    public static List<IMSummarizedStudent> getSummarizedStudents
        (List<IMStudent> students) {
        List<IMSummarizedStudent> result =
            new ArrayList<IMSummarizedStudent>();
        for (IMStudent $_iter:students)
            result.add(getSummarizedStudent($_iter));
        return result;
    }

    public static IMSummarizedStudent getSummarizedStudent
        (IMStudent student) {
        if (MSSStudentToAccept.checkStudentToAccept(student))
            return new MSummarizedStudent(MSAcceptStudent.
                getAcceptedStudent(student));
        return new MSummarizedStudent(MSFailStudent.
            getFailedStudent(student));
    }
}

```

Fig. 15. Code for determining student's eligibility for promotion to the next semester

```

public class ServiceImpl implements IService {
    // ...
    List<SemesterSummarizationDataItemDTO>
        preparesSemesterSummarizationData
            (List<StudentListItemDTO> studentListDTO)
    {
        return MSummarizeStudentAfterSemester.
            getSummarizedStudents(studentListDTO);
    }
    // ...
}

```

Fig. 16. Additional back-end access code

according to rule no. 15 (generate code from transition patterns).

Fig. 15 shows code generated on the basis of the model from Fig. 12. This time we can see only one method ("getSummarizedStudent") generated according to rule no. 9. The method's body is generated on the basis of a 'mapping' transition pattern with the formula 'studentToAccept(student); acceptStudent(student); failStudent(student)'. Formulas for this type of transition patterns are composed of three sections: a 'simple' condition pattern formula and two 'simple' transition pattern formulas (the first one used when the condition is true, and the second one otherwise). Here, the condition pattern refers to the 'student to accept' relationship shown in Fig. 10. Thus, the current method calls the 'checkStudentToAccept' method shown in Fig. 13. Depending on its result, it calls one of two methods resulting from transforming the 'accept student' and 'fail student' relationships. The other method in this class invokes the above described one over a set of appropriate objects (list of students).

Finally, relevant code fragments as presented above, can be now applied to fill-in appropriate empty methods of the back-end service class (see Fig. 3). In our example this pertains to the method that corresponds to a non-CRUD operation. Appropriate additional code is presented in Fig. 16. It contains a simple call to the operation presented in Fig. 15. It is worth noting that such operations are optimised to use only required parameters.

VII. LANGUAGE VALIDATION AND DISCUSSION

To initially validate the presented language, we have used two different approaches. Their aim was to assess certain aspects of the language's usability: understandability and operability. The first approach was to determine language comprehension by its first-time users. It consisted in testing language proficiency, following a brief introduction to the language. The second approach was to determine efficiency of language usage by the users with various experience levels. In both cases, we have used a specially developed RSL-DL editor, used in conjunction with the ReDSeeDS environment.

A. Validation of understandability

The first validation study was conducted with a group of post-graduate computer science students attending the "Model-Driven Software Development" course at the Warsaw University of Technology. The course curriculum included classes on the design and usage of various Domain-Specific Languages. The study was thus well aligned with the aim to acquaint the students with this topic.

The setup of the study was as follows. First, the students attended two lab sessions (four class hours) where they were presented with the RSL and the ReDSeeDS tool. Note that prior to this, the students had no experience with Software Language Engineering but have attended a parallel lecture where they were introduced with the fundamentals of meta-modelling. Next, the students were presented with a brief, one-hour introduction to the language. Then, they have spent two hours solving simple exercises using the aforementioned RSL-DL editor. After this, the students were presented with correct solutions to the exercises. Finally, the students were asked to answer 12 questions in an online questionnaire. All of the questions were single-choice, and referred to specific RSL-DL diagrams. Each question had four possible answers. The first eight questions were related to the understanding of the language syntax, the next three related to language usage, and the final one checked more nuanced usage of the language related to its declarative nature. The students were given one class hour (45 minutes) to finish the questionnaire, but most of them have finished in less than 20 minutes.

The results of the study consist in 42 replies to the questionnaire. The average of correct answers in the whole questionnaire was 69%. For syntax understanding (the first eight questions), it was 75%, for usage understanding (the next three questions), it was 60%, and for the last question it was 40%. Detailed results, together with the question contents are provided in the Supplement.

The relatively low result in the case of the last question can be explained by its advanced nature, going beyond the explanations given to the students. Thus, the 40% can be seen as an unexpectedly good result. It is also worth noting that relatively low percentages of correct answers were associated with questions about differentiation between inferred roles and assigned roles (questions no. 3, 5 and 9). These results were similar to the case of the last question. Further research is

TABLE I
RESULTS OF THE OPERABILITY STUDY

Participant	Task	RSL-DL time	Java time
Author	No. 1	3:00 min.	5:00 min.
Ph.D. Student	No. 1	9:00 min	12:00 min.
Undergrad. Student	No. 1	13:35 min	4:50 min.
Author	No. 2	1:45 min.	3:40 min.
Ph.D. Student	No. 2	4:00 min	10:00 min.
Undergrad. Student	No. 2	16:45 min	5:30 min.
Author	No. 3	4:30 min.	6:00 min.
Ph.D. Student	No. 3	15:00 min	15:00 min.
Undergrad. Student	No. 3	12:25 min	6:20 min.

needed to determine if that was caused by insufficient explanations (this aspect was under-represented in the exercises) or inherent difficulties caused by the language design. In summary, the overall results of this study indicate that the language is comprehensible even after a very short introduction. However, a more thorough validation with statistical analysis is needed to confirm this, and can be seen as future work.

B. Validation of operability

The second validation study was conducted with a group of three software developers with different programming skills. The first person is one of the language authors and thus has very good knowledge of RSL-DL. At the same time, he is an experienced Java programmer. The second person is a Ph.D. student with wide general computer science knowledge and average Java programming experience. The third person is an undergraduate student with more narrow CS knowledge but with relatively high experience in Java programming. The students were not involved in the development of RSL-DL and had no previous knowledge of it.

The study consisted in comparison of coding efficiency and was based on solving specific problems. The setup of the study was as follows. First, the study participants were presented with a 1.5 hour long introduction of RSL-DL and its editor. This included the presentation of three problems: calculation of square mean error (no. 1), calculation of definite integrals (no. 2), and calculation of VAT for product lists (no. 3). The problem formulations involved appropriate formulas and are presented in detail in the Supplement. Next, the participants were supplied with artefacts generated from appropriate RSL specifications (use cases with scenarios) by the ReDSeeDS system. These consisted of pre-initialised RSL-DL models (just the notions) and code skeletons (Data Transfer Objects and method signatures) in Java.

The goal of the participants was to fill-in the provided artefacts to complete domain logic functionality. To prevent from negative bias, the participants were asked to solve the problems using RSL-DL first, and only then to solve them in Java. The participants were also asked to measure time spent on all the tasks. The results of these measurements are given in Table I.

Comparison of times for the three study participants can be treated as rather anecdotal evidence but they give some insight on the productivity of developing domain logic (backend) code

with RSL-DL. As it can be noticed, productivity of RSL-DL development vs. Java development is significantly higher for an experienced RSL-DL user. Also, a less experienced Java programmer (the Ph.D. student) had certain productivity gains. On the other hand, a very experienced Java programmer (the undergraduate student) had performed much better using a traditional programming language. Thus, it can be argued that as general knowledge of developers and their RSL-DL skills raise - productivity gains tend to be significant. It can also be argued that RSL-DL has the potential for extending productivity gains for less experienced programmers. Still, this argumentation has to be acknowledged through a more thorough experimentation with a larger participant scope. This can be seen as future work.

VIII. SUMMARY AND FUTURE WORK

In this paper, we have shown that declarative knowledge representations can be used to produce imperative (3GL) backend code with non-trivial domain logic. Moreover, this code can be interfaced with front-end code produced from low-code specifications that use formalised requirements models (use cases, scenarios). To achieve this, we have used a combination of techniques from model-driven development and ontology-based inference. In this environment, most “programming” activities could be made using a high-level visual language. Thus, programming becomes equivalent to specifying models that define various aspects of the system and its problem domain. An RSL-DL conceptual model can be treated – in fact – as a high-level program that can be executed immediately after compiling it into eg. Java and then – executable code. Moreover, RSL-DL models can be seen as “ontologies as code” [19] and a step towards a “fifth generation language” as postulated by Thalheim and Jaakkola [35].

We see two main areas where our approach can benefit software development: reduction of complexity and increased reuse. The first area is in line with the general goals of the low-code movement – to offer means for reducing accidental (technological) complexity in favour of concentrating on the essential (e.g. domain) complexity (see early insights on this by Brooks [6]). A thorough comparison of complexity between RSL-DL and traditional programming languages, and analysis of reusability can be seen as interesting areas of future work.

Another area for future work is the analysis of RLS-DL usability as a low-code language. Generally, it can be expected that better usability is assured through declarative characteristics of the language (see appropriate comparative analyses [15], [37]). This is in line with our initial studies presented in the previous section. However, to fully support this claim, more extensive experimentation should be conducted.

Other areas which we plan to investigate in the future include better integration with requirements processing mechanisms. One aspect of this is the application of natural language processing. Here, valuable insights can be drawn from the concept of naturalistic programming [24]. This concept postulates the use of natural language elements to design programming languages that are more expressive from the programmers’

point of view. Another interesting approach in this area is that of Mefteh et al. [20]. In this approach, natural language scenarios are transformed into constrained language models expressed in RSL. On the other hand, we also plan to integrate our approach with existing approaches to generate CRUD operations and database schemas directly from requirements models expressed in RSL [39].

As a final remark we address the question of creating a distinct new language as opposed to using an extension to the existing language (e.g. a UML profile). In our opinion, from the practical point of view, both approaches can be seen as equivalent. However, creating a language from scratch favours solutions that go beyond the “beaten path”.

REFERENCES

- [1] *The MOLA Language Reference Manual Version 2.0 final*, 2007.
- [2] Colin Atkinson, Matthias Gutheil, and Kilian Kiko. On the relationship of ontologies and models. In *Proc. 2nd Workshop on Meta-Modelling, WoMM 2006*, pages 47–60, 2006.
- [3] Fábio Paulo Basso, Raquel Mainardi Pillat, Toacy Cavalcante Oliveira, Fabricia Roos-Frantz, and Rafael Z. Frantz. Automated design of multi-layered web information systems. *Journal of Systems and Software*, 117:612–637, 2016.
- [4] Marco Brambilla and Piero Fraternali. *Interaction flow modeling language: Model-driven UI engineering of web and mobile apps with IFML*. Morgan Kaufmann, 2014.
- [5] Isabel Sofia Brito, Joao Paulo Barros, and Luis Gomes. From requirements to code (Re2Code) – a model-based approach for controller implementation. In *2016 IEEE 14th International Conference on Industrial Informatics (INDIN)*, pages 1224–1230, 2016.
- [6] Frederick P Brooks. No silver bullet: Essence and accidents of software engineering. *IEEE Computer*, 20(4):10–19, April 1987.
- [7] Jordi Cabot. Positioning of the low-code movement within the field of model-driven engineering. In *Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings, MODELS '20*, 2020.
- [8] Malcolm Chisholm. *How to build a business rules engine: extending application functionality through metadata engineering*. Morgan Kaufmann, 2004.
- [9] Dragan Gašević, Dragan Djurić, and Vladan Devedžić. *Model Driven Engineering and Ontology Development*. Springer, 2009.
- [10] Hele Mai Haav and Andres Ojamaa. Semi-automated integration of domain ontologies to DSL meta-models. *International Journal of Intelligent Information and Database Systems*, 10(1/2):94–116, 2017.
- [11] Brian Henderson-Sellers. Bridging metamodels and ontologies in software engineering. *Journal of Systems and Software*, 84(2):301–313, 2011.
- [12] Martin Henkel and Janis Stirna. Pondering on the key functionality of model driven development tools: The case of Mendix. In *International Conference on Business Informatics Research*, pages 146–160. Springer, 2010.
- [13] Henrique Henriques, Hugo Lourenço, Vasco Amaral, and Miguel Goulão. Improving the developer experience with a low-code process modelling language. In *21th ACM/IEEE International Conference on Model Driven Engineering Languages and Systems*, pages 200–210, 2018.
- [14] Michael G Hinchey, James L Rash, and Christopher A Rouff. Requirements to design to code: Towards a fully formal approach to automatic code generation. Technical report, NASA, 2005.
- [15] Ahmad Jbara, Arieh Bibliowicz, Niva Wengrowicz, Natali Levi, and Dov Dori. Toward integrating systems engineering with software engineering through object-process programming. *International Journal of Information Technology*, pages 1–35, 2020.
- [16] Audris Kalnins, Janis Barzdins, and Edgars Celms. Model transformation language MOLA. *Lecture Notes in Computer Science*, 3599:62–76, 2004.
- [17] Nora Koch, Santiago Meliá-Beigbeder, Nathalie Moreno-Vergara, Vicente Pelechano-Ferragud, Fernando Sánchez-Figueroa, and J Vara-Mesa. Model-driven web engineering. *Upgrade-Novática Journal (English and Spanish)*, 2:40–45, 2008.
- [18] Axel Kramer. Symja library-Java symbolic math system, 2018. last accessed May 2020.
- [19] Beatriz Franco Martins. The OntoOO-method: An ontology-driven conceptual modeling approach for evolving the oo-method. In *Advances in Conceptual Modeling*, pages 247–254, 2019.
- [20] Mariem Mefteh, Nadia Bouassida, and Hanene Ben-Abdallah. Towards naturalistic programming: Mapping language-independent requirements to constrained language specifications. *Science of Computer Programming*, 166:89–119, 2018.
- [21] Wiktor Nowakowski, Michał Śmiałek, Albert Ambroziewicz, and Tomasz Straszak. Requirements-level language and tools for capturing software system essence. *Computer Science and Information Systems*, 10(4):1499–1524, 2013.
- [22] Object Management Group. *Meta Object Facility (MOF) Core Specification, version 2.5.1, formal/2019-10-01*, 2019.
- [23] Fernando Silva Parreiras, Steffen Staab, Simon Schenk, and Andreas Winter. Model driven specification of ontology translations. In *ER'08, volume 5231 of Lecture Notes in Computer Science*, pages 484–497, 2008.
- [24] Oscar Pulido-Prieto and Ulises Juárez-Martínez. A survey of naturalistic programming technologies. *ACM Comput. Surv.*, 50(5), 2017.
- [25] Clay Richardson, John R Rymer, Christopher Mines, Alex Cullen, and Dominique Whittaker. New development platforms emerge for customer-facing applications, 2014. Forrester report.
- [26] Roberto Rodriguez-Echeverria, Juan C Preciado, Alvaro Rubio-Largo, José M Conejero, and Alvaro E Prieto. A pattern-based development approach for Interaction Flow Modeling Language. *Scientific Programming*, 2019, 2019.
- [27] Davide Di Ruscio, Dimitris Kolovos, Juan de Lara, Alfonso Pierantonio, Massimo Tisi, and Manuel Wimmer. Low-code development and model-driven engineering: Two sides of the same coin? *Software and Systems Modeling*, pages 1–10, jan 2022.
- [28] Kamil Rybiński and Rafal Parol. RSL-DL: Representing domain knowledge for the purpose of code generation. In *Software Engineering: Challenges and Solutions*, pages 61–73. Springer, 2017.
- [29] A. Sahay, A. Indamutsa, D. Di Ruscio, and A. Pierantonio. Supporting the understanding and comparison of low-code development platforms. In *46th Euromicro Conference on Software Engineering and Advanced Applications*, pages 171–178, 2020.
- [30] Kenneth Slonneger and Barry L Kurtz. *Formal Syntax and Semantics of Programming Languages*. Addison-Wesley, 1995.
- [31] Michał Śmiałek, Norbert Jarzebowski, and Wiktor Nowakowski. Runtime semantics of use case stories. In *Visual Languages and Human-Centric Computing (VL/HCC), 2012 IEEE Symposium on*, pages 159–162. IEEE, 2012.
- [32] Michał Śmiałek and Wiktor Nowakowski. *From Requirements to Java in a Snap: Model-Driven Requirements Engineering in Practice*. Springer, 2015.
- [33] Michał Śmiałek and Tomasz Straszak. Facilitating transition from requirements to code with the ReDSeeDS tool. In *Requirements Engineering Conference (RE), 2012 20th IEEE International*, pages 321–322, 2012.
- [34] Graeme Stevenson and Simon Dobson. Sapphire: Generating java runtime artefacts from OWL ontologies. In *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, pages 425–436. Springer, 2011.
- [35] Bernhard Thalheim and Hannu Jaakkola. Model-based fifth generation programming. In *Information Modelling and Knowledge Bases XXXI*, pages 381–400. IOS Press, 2020.
- [36] Max Völkel and York Sure. RDFReactor-from ontologies to programmatic data access. In *Poster Proceedings of the Fourth International Semantic Web Conference*, 2005.
- [37] Petri Vuorimaa, Markku Laine, Evgenia Litvinova, and Denis Shestakov. Leveraging declarative languages in web application development. *World Wide Web*, 19(4):519–543, 2016.
- [38] Karzan Wakil and Dayang NA Jawawi. Model driven web engineering: A systematic mapping study. *e-Informatica Software Engineering Journal*, 9(1):107–142, 2015.
- [39] Nassima Yamouni Khelifi, Michał Śmiałek, and Rachida Mekki. Generating database access code from domain models. In *2015 Federated Conference on Computer Science and Information Systems*, pages 991–996, 2015.

7th Doctoral Symposium on Recent Advances in Information Technology

THE aim of this meeting is to provide a platform for exchange of ideas between early-stage researchers, in Computer Science and Information Systems, PhD students in particular. Furthermore, the symposium will provide all participants an opportunity to get feedback on their studies from experienced members of the IT research community invited to chair all DS-RAIT thematic sessions. Therefore, submission of research proposals with limited preliminary results is strongly encouraged.

Besides receiving specific advice for their contributions all participants will be invited to attend plenary lectures on conducting high-quality research studies, excellence in scientific writing and issues related to intellectual property in IT research. Authors of the two most outstanding submissions will have a possibility to present their papers in a form of short plenary lecture.

TOPICS

- Automatic Control and Robotics
- Bioinformatics
- Cloud, GPU and Parallel Computing
- Cognitive Science
- Computer Networks
- Computational Intelligence
- Cryptography
- Data Mining and Data Visualization
- Database Management Systems
- Expert Systems
- Image Processing and Computer Animation
- Information Theory
- Machine Learning
- Natural Language Processing
- Numerical Analysis
- Operating Systems
- Pattern Recognition
- Scientific Computing
- Software Engineering

TRACK CHAIRS

- **Kowalski, Piotr**, Systems Research Institute, Polish Academy of Sciences; AGH University of Science and Technology, Poland
- **Łukasik, Szymon**, Systems Research Institute, Polish Academy of Sciences, AGH University of Science and Technology, Poland

PROGRAM COMMITTEE

- **Arabas, Jaroslaw**, Warsaw University of Technology, Poland
- **Atanassov, Krassimir T.**, Bulgarian Academy of Sciences, Bulgaria
- **Balazs, Krisztian**, Budapest University of Technology and Economics, Hungary
- **Bronselaeer, Antoon**, Department of Telecommunications and Information at Ghent University, Belgium
- **Castrillon-Santana, Modesto**, University of Las Palmas de Gran Canaria, Spain
- **Charytanowicz, Malgorzata**, Catholic University of Lublin, Poland
- **Corpetti, Thomas**, University of Rennes, France
- **Courty, Nicolas**, University of Bretagne Sud, France
- **De Tré, Guy**, Faculty of Engineering and Architecture at Ghent University, Belgium
- **Fonseca, José Manuel**, UNINOVA, Portugal
- **Fournier-Viger, Philippe**, University of Moncton, Canada
- **Gil, David**, University of Alicante, Spain
- **Herrera Viedma, Enrique**, University of Granada, Spain
- **Hu, Bao-Gang**, Institute of Automation, Chinese Academy of Sciences, China
- **Koczy, Laszlo**, Szechenyi Istvan University, Hungary
- **Kokosinski, Zbigniew**, Cracow University of Technology, Poland
- **Krawiec, Krzysztof**, Poznan University of Technology, Poland
- **Kulczycki, Piotr**, Systems Research Institute, Polish Academy of Sciences, Poland
- **Kusy, Maciej**, Rzeszow University of Technology, Poland
- **Lilik, Ferenc**, Szechenyi Istvan University, Hungary
- **Lovassy, Rita**, Obuda University, Hungary
- **Malecki, Piotr**, Institute of Nuclear Physics PAN, Poland
- **Mesiar, Radko**, Slovak University of Technology, Slovakia
- **Mora, André Damas**, UNINOVA, Portugal
- **Noguera i Clofent, Carles**, Institute of Information Theory and Automation (UTIA), Academy of Sciences of the Czech Republic, Czech Republic
- **Pamin, Jerzy**, Institute for Computational Civil Engineering, Cracow University of Technology, Poland
- **Petrik, Milan**, Czech University of Life Sciences Prague, Faculty of Engineering, Department of Mathematics, Czech Republic

- **Sachenko, Anatoly**, Ternopil State Economic University, Ukraine
- **Samotyy, Volodymyr**, Lviv State University of Life Safety, Ukraine
- **Szafran, Bartlomiej**, Faculty of Physics and Applied Computer Science, AGH University of Science and Technology, Poland
- **Tormasi, Alex**, Szechenyi Istvan University, Hungary
- **Wei, Wei**, School of Computer science and engineering, Xi'an University of Technology, China
- **Wysocki, Marian**, Rzeszow University of Technology, Poland
- **Yang, Yujiu**, Tsinghua University, China
- **Zadrozny, Slawomir**, Systems Research Institute, Poland
- **Zajac, Mieczyslaw**, Cracow University of Technology, Poland

Impact of clustering of unlabeled data on classification: case study in bipolar disorder

Olga Kamińska, Katarzyna Kaczmarek-Majer, Olgierd Hryniewicz
Systems Research Institute Polish Academy of Sciences
ul. Newelska 6, 04-710 Warsaw, Poland
Email: {okaminska, k.kaczmarek, o.hryniewicz}@ibspan.waw.pl

Abstract—Currently, it is possible to collect a large amount of data from sensors. At the same time, data are often only partially labeled. For example, in the context of smartphone-based monitoring of mental state, there are much more data collected from smartphones than those collected from psychiatrists about the mental state. The approach presented in this paper is designed to examine if unlabeled data can improve the accuracy of classification tasks in the considered case study of classifying a patient’s state. First, unlabeled data are represented by clusters membership through Fuzzy C-means algorithm which corresponds to the uncertainty of the patient’s condition in this disease. Secondly, the classification is performed using two well-known algorithms, Random Forest and SVM. The obtained results indicate a minimal improvement in the quality of classification thanks to the use of membership in clusters. These results are promising due to both, the accuracy and interpretability.

I. INTRODUCTION

MOTIVATION for this research comes from the practical problem of classifying partially labeled data. Within this work, we concentrate on a particular case study in monitoring the mental state of bipolar disorder (BD) patients which has a large dataset of sensor-based data with labels provided by doctors. Since we are limited by medical labels, the most frequent attempts to predict a patient’s condition come down to using only a small part of the data from the entire set. Such selected data may not contain characteristics for each patient and the obtained results may not be accurate. To alleviate the aforementioned problem, we propose to incorporate the results of clustering into the classification task.

The collection of medical data in our possession indicates to use of a semi-supervised approach. For this purpose, it is worth enabling clustering to extract information from unlabeled data [1]. In related work, the accuracy of classification for patients with bipolar disorder using sensor data amounts to 76% [2]. Some works test the influence of clusters in the classification problems, e.g., [3] and indicate an improvement in the results. Other works include unlabeled data by means of the dynamic incremental fuzzy semi-supervised learning, see e.g., [4].

Experiments are performed on data about voice collected from smartphones of bipolar disorder patients. On the other hand, labels obtained from psychiatrists during visits are valid only for daytime, and of that day only so the amount of those labels is relatively small. Psychiatrists indicate that the symptoms of a given phase are visible several days before the visit, therefore the validity of this label can be extended. That

procedure results in the preparation of a much larger range of data enabling the classification of the patient’s condition. Therefore, we check whether the use of the remaining unlabeled data represented by the membership to clusters improves the quality of the classification labeled data.

II. METHODOLOGY

The idea of the proposed experiment is to verify that unlabeled data assign as a clusters membership has an impact on the classification of patients states.

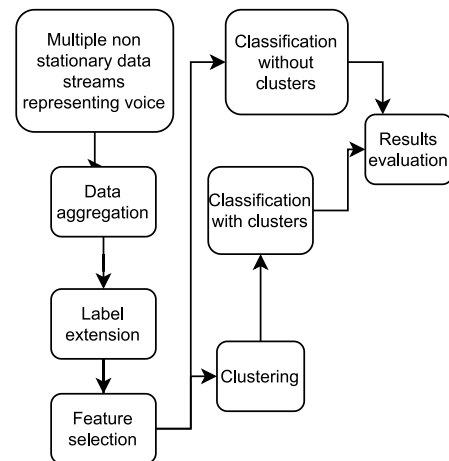


Fig. 1. Process flow

The process flow of the proposed experiment is presented in Figure 1 and in Algorithm 1. The experiment begins with retrieving all available multiple non-stationary data streams representing voice. All frames are then aggregated to the patient’s phone call level with different aggregates methods. Psychiatric assessments obtained during visits, represented as labels are spread around the day of the patient’s visit. Within the feature selection, the top-k most important voice parameters are selected for each patient and aggregated methods. Additionally, all available patient data are clustered to include unlabeled data as well. Simultaneously, the classification of the patient’s condition is carried out on the data containing clusters membership and without this information. Finally, results are evaluated with multiple metrics.

Algorithm 1 Pseudocode of the experiment

```

1 for patient in patients_list:
2   for agg in aggregates:
3     best_features = RFE(no_of_best_parameters = 10)
4     cluster_mmbs = FuzzyCmeans(data[, best_features], unique_visits_no)
5     rf_nocluster = RandomForest(data_without_clusters)
6     rf_withcluster = RandomForest(data_with_clusters)
7     SVM_nocluster = SVM(data_without_clusters)
8     SVM_withcluster = SVM(data_with_clusters)

```

A. Data collection and aggregation

Patients were enrolled and used a dedicated smartphone application in everyday life starting in September 2017 and ending in December 2018. All their recordings were divided into 10-20 ms frames. Next, for those frames, openSMILE [5] library was used to calculate acoustic features. The final dataset contains 86 data streams that describe the main acoustic features of voice such as e.g., loudness, voice energy, pitch, etc. All data from 2018 were selected for the experiment for each of the 6 patients who had several visits in the year of the study where different disease phases were observed.

Due to a large number of frames available for each connection, this data has been aggregated to the level of one phone call using mean, standard deviation, skewness and quartiles. Each of the available 85 voice parameters is aggregated in this way. The data were then normalized. Aggregating the data to the level of the conversation will allow you to slightly reduce the noise of the data and facilitate data processing in subsequent processes.

B. Labels extension

The labels obtained by the patient during the visit are valid only on the day of the visit. The number of labels available for a given patient during the year did not exceed 7 for 1 patient, which is a negligible value throughout the year. The ability to extend the label to days around the visit increases the amount of labeled data. Other studies are considering extending the label to 7 days in advance as symptoms of the patient's condition may already be noticeable prior to the visit. In the present experiment, the label was extended 7 days before the visit and 2 days after the visit. This gives the label a validity period of 10 days. Results received from that method are shown in TABLE I. In an example of first patients with ID 1472 we can observe, that label extension increased the number of phone calls with labels from 42 to 391. There was much more unlabeled data, i.e. 1482, what is worth using it.

C. Feature Selection

To obtain significant voice parameters we apply one of the automatic feature selection methods called Recursive Feature Elimination (RFE) [6]. The idea of the RFE technique is to build a model with all variables and after that, the algorithm successfully removes features until the desired number remains. The current implementation of that method used the Random Forest algorithm to train and create ranking features by importance, discarding the least important features, and

TABLE I
PATIENTS DETAILS

ID patient	No. of visits	No of phone calls in day of visits	No of phone calls for extended label validity	No of all phone calls in 2018
1472	2	42	391	1482
2004	2	16	223	871
2582	3	31	169	645
5656	2	7	29	215
5736	2	20	71	1025
6139	3	22	90	254

re-fitting the model. To find the optimal number of features cross-validation is used with the RFE algorithm to obtain the best scoring collection of features. The final subset used the first 10 best parameters resulting from that method.

Earlier studies [7] have shown that the introduction of the RFE method improves clustering results. In the present work, the RFE method is used for each patient and each data aggregation method separately considering only the labeled data. This allows the best voice parameters to be selected in a tailored way. This set of the 10 most important acoustic parameters is then used in the next stage of the experiment.

D. Clustering

The algorithm used for clustering is Fuzzy C-mean [8] with squared Euclidean distances as a parameter of dissimilarities between observations. We assume a patient may have symptoms of several conditions at the same time during unlabeled days. This happens mainly in a mixed state where the symptoms of mania and depression occur simultaneously. Furthermore, on unlabeled days the patient may not have obvious symptoms characteristic of only one BD state. Such uncertainty resulted in the choice of the Fuzzy C-mean algorithm which introduces cluster membership. In that case, the number of clusters corresponds to the known number of different phases diagnosed in a given patient.

In fuzzy clustering, each observation is "spread out" over the various clusters. The output of that clustering is the membership to each of the clusters. The memberships are nonnegative, and for a fixed observation it sums to 1. So each phone call could be partly assigned to one class and partly to another class. We don't assign a specific cluster to BD state. We just looking for a variety between classes.

Clustering was performed on all available data for each patient (also unlabeled data) in order to capture the variability over all available observations. These data were aggregated

to the level of the patient's conversation and then the 10 most important acoustic parameters were selected by the RFE method.

E. Classification

The classification was made in 2 ways using 2 well-known classifiers, the Random Forest [9] and Support Vector Machine [10]. The first method (lines 5 and 7 in Algorithm 1) assumes that aggregated data with selected voice parameters by the RFE method are classified. The second method (lines 6 and 8 in Algorithm 1) additionally joined the membership of each cluster to that dataset. Clusters membership were used for labeled data only. Then the data is divided into a training set and a test set in the proportion of 75:25 in such a way that each set contains data from each BD patient's state. Predicted classes depend on how many different phases the patient received during the whole study. The classification is performed on each patient for each of the 6 available aggregation methods. The "best aggregate" is then selected according to the Accuracy comparison of each aggregate for that patient. The classification results for the selected aggregate are summed up from all patients and a common confusion matrix is created for the selected algorithm and data without clusters and with clusters. The values in the confusion matrix are presented for the test set not used during training.

F. Evaluation metrics

In total 4 confusion matrices have been created. The first two matrices concern the comparison of values from all patients and their best aggregation methods for the Random Forest algorithm in the first case without the use of clusters and in the second additionally including cluster membership.

The next 2 matrices also compare the values without clusters and with clusters, this time using the SVM algorithm.

Additionally, for each of the above-mentioned matrices, classification coefficients were calculated, such as Accuracy - correctly forecasted patient states concerning all forecasts, Precision - i.e. the fraction of relevant instances among the retrieved instances, and Recall - i.e. the fraction of relevant instances that were retrieved.

III. EXPERIMENTAL RESULTS

A. Selection of aggregation operators and acoustic features

Selecting an appropriate aggregation operator for the acoustic data is not obvious, therefore, several such methods were tested in this study

The best aggregates were selected separately for each patient. The results are presented in TABLE II. The best aggregating methods turned out to be the mean and skewness. They have been selected 7 times. Then Q1 was selected 5 times, Q3 - 3 times, and Standard deviation - 2 times. Interestingly, the Q2 aggregate known as the median was not selected even once. Moreover, the parameters that best characterize the normal distribution, i.e. mean, standard deviation, and skewness, were selected twice as often (16 hits) as the parameters characterizing the quantile distribution (8 hits). The

differences are also noticeable concerning the classifiers used. However, it does not affect their further interpretability.

The relevant acoustic parameters received from RFE methods differ for each model as well. The following 10 parameters were most often selected by the model are: *spectral-RollOff90*, *LOGenergy*, *mfcc 11*, *fband1000-4000*, *f3frequency*, *f2frequency*, *fband0-650*, *hammarbergindex*, *audSpec*, *spectralharmonicity*.

TABLE II
AGGREGATES METHODS SELECTION

	RF non-cluster	RF with cluster	SVM non-cluster	SVM with cluster
<i>aggregate</i>	<i>cardinality</i>			
mean	1	1	2	3
standard deviation	1	0	1	0
skewness	2	2	2	1
Q1	1	2	1	1
Q2	0	0	0	0
Q3	1	1	0	1

B. Classification

1) *Random Forest*: Table III contains the confusion matrices for the test sets from all patients where the data included in the Random Forest classifier did not include cluster membership (left) and where the data contained cluster membership (right). Values on the diagonal of the matrix indicate the correct classification of each of the states. The remaining values indicate what were the forecasts for the remaining cases where the observed label does not agree with the predicted value. The results are promising for both models. The total number of correctly classified for the non-clustering model is 209 and for the clustering model is 211, so we see a slight improvement in the results. The only place where clusters join in shows a slight weakening of the results is in the prediction of the state of depression. The other values are slightly better or equal. However, these differences are subtle, so it should be tested on more examples.

Table IV contains the results calculated based on the above-mentioned confusion matrices. As observed, the precision and recall values for each state are similar or slightly better for the method containing cluster memberships. Accuracy, i.e. the ratio of correctly predicted values to all values, also increases in the method containing clusters from 83.27% to 84.06%.

2) *SVM*: Table III contains the confusion matrix summed for the test set from all patients where the data included in the Support Vector Machine classifier did not include cluster membership (left) and where the data contained cluster membership (right). Results obtained for the SVM are similar to the previously considered RF. The total number of correctly classified for the non-clustering model is 203 and for the clustering model is 210, so there is again a slight improvement. In that case, there is no place where the method using clusters received a worse number of incorrect predicted values in any class.

TABLE III
CONFUSION MATRICES FOR RF AND SVM CLASSIFIERS WITHOUT (LEFT) AND WITH INFORMATION ABOUT MEMBERSHIP TO CLUSTERS.

RF-nc		actual			
		E	X	D	M
predicted	E	79	15	13	1
	X	4	76	0	0
	D	9	0	39	0
	M	0	0	0	15

RF-c		actual			
		E	X	D	M
predicted	E	83	11	13	1
	X	3	77	0	0
	D	12	0	36	0
	M	0	0	0	15

SVM-nc		actual			
		E	X	D	M
predicted	E	79	17	11	1
	X	2	78	0	0
	D	17	0	31	0
	M	0	0	0	15

SVM-c		actual			
		E	X	D	M
predicted	E	82	17	13	1
	X	2	78	0	0
	D	15	0	35	0
	M	0	0	0	15

TABLE IV
RESULTS OF RF AND SVM CLASSIFIERS WITHOUT (LEFT) AND WITH CLUSTERS(RIGHT)

RF-nc	E	X	D	M
precision (PPV)	0.73	0.95	0.81	1
recall (TPR)	0.86	0.84	0.74	0.94
accuracy	83.27%			

SVM-nc	E	X	D	M
precision (PPV)	0.73	0.98	0.65	1
recall (TPR)	0.80	0.82	0.74	0.94
accuracy	80.88%			

RF-c	E	X	D	M
precision (PPV)	0.78	0.96	0.75	1
recall (TPR)	0.85	0.88	0.74	0.94
accuracy	84.06%			

SVM-c	E	X	D	M
precision (PPV)	0.73	0.98	0.70	1
recall (TPR)	0.83	0.82	0.73	0.94
accuracy	81.40%			

Table IV contains the results calculated based on the above-mentioned Confusion Matrix. Values received by that classifier are similar to the previous one. Both precision and recall values for each state are similar or slightly better for the method containing cluster memberships. Accuracy, i.e. the ratio of correctly predicted values to all values, also increases in the method containing clusters from 80.88% to 81.40%.

Results obtained by both classifiers are promising. Both classifiers achieved high precision and recall that indicate the correctly predicted class. Moreover, joining a cluster membership has a slightly better effect on the quality of the classification.

IV. CONCLUSION

Unlabeled data appear to have a positive effect on the accuracy of classifying patients with bipolar disorder. In the study, a model was prepared that fits each patient individually. The presented solution individually sets the list of important acoustic parameters, the appropriate method of aggregating these data, and the number of clusters in which the patient may be. Such a model was compared with a model that did not include membership of clusters (so used amount of data from each patient was used for the model). The presented results indicate that adding information about clusters slightly improves the classification performance. The current results seem to be promising, and the study will be repeated for the remaining patients.

ACKNOWLEDGMENT

Olga Kamińska and Katarzyna Kaczmarek-Majer are supported by the Small Grants Scheme

(NOR/SGS/BIPOLAR/0239/2020-00) within the research project: "Bipolar disorder prediction with sensor-based semi-supervised Learning (BIPOLAR)".

REFERENCES

- [1] A. Bouchachia and W. Pedrycz, "Data clustering with partial supervision," *Data Min. Knowl. Discov.*, vol. 12, no. 1, p. 47–78, jan 2006. [Online]. Available: <https://doi.org/10.1007/s10618-005-0019-1>
- [2] A. Grünerbl, A. Muaremi, and V. Osmani, "Smartphone-based recognition of states and state changes in bipolar disorder patients," *IEEE Journal of Biomedical and Health Informatics*, vol. 19(1), 2015.
- [3] T. Chakraborty, "Ec3: Combining clustering and classification for ensemble learning," in *2017 IEEE International Conference on Data Mining (ICDM)*, 2017, pp. 781–786.
- [4] G. Casalino, G. Castellano, F. Galetta, and K. Kaczmarek-Majer, "Dynamic incremental semi-supervised fuzzy clustering for bipolar disorder episode prediction," in *Discovery Science. DS 2020*, A. Appice and et al., Eds., 2020.
- [5] F. Eyben, F. Wengler, F. Gross, and B. Schuller, "Recent developments in opensmile, the munich open-source multimedia feature extractor," in *Proc. of the 21st ACM Int. Conf. on Multimedia*, 2013, pp. 835–838.
- [6] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Machine learning*, vol. 46, no. 1-3, pp. 389–422, 2002.
- [7] O. Kamińska, K. Kaczmarek-Majer, and O. Hryniewicz, "Acoustic feature selection with fuzzy clustering, self organizing maps and psychiatric assessments," *Proceedings of Information Processing and Management of Uncertainty in Knowledge-Based Systems, IPMU 2020, Lisbon*, 2020.
- [8] J. Bezdek, R. Ehrlich, and W. Full, "Fcm: Fuzzy c-means algorithm," 1984.
- [9] M. Pal, "Random forest classifier for remote sensing classification," *Int. J. Remote Sens.*, pp. 217–222, 2005.
- [10] K. Srinivasan, N. Mahendran, D. R. Vincent, C.-Y. Chang, and S. Syed-Abdul, "Realizing an integrated multistage support vector machine model for augmented recognition of unipolar depression," *Electronics*, vol. 9, no. 4, p. 647, 2020.

Parameters Estimation of a Lotka-Volterra Model in an Application for Market Graphics Processing Units

Dzhakhongir Normatov and Paolo Mercorelli
 Institute of Product and Process Innovation
 Leuphana University of Lüneburg
 Universitaetsallee 1, 21335 Lüneburg
 Germany

Abstract—In this paper, a least squares method is used to estimate parameter values in the Lotka-Volterra model. The data used are graphics processing units (GPU) shipment worldwide by three key competitors, namely Nvidia, Intel, AMD. The goal is to quantify the parameter values of a model with minimal error in order to qualitatively solve the problem and fit the raw data as closely as possible. Based on the real measurements, the predator between the competitors is recognized through the identification procedure comparing the sign of the coefficients with the original Lotka-Volterra model structure.

I. INTRODUCTION

OUR NATURAL environment is mostly in motion. In order to visualize and model the ever-changing natural environment and, more specifically, population dynamics under a variety of circumstances, the Lotka-Volterra equations have proven to be a useful tool, as long as we build our investigations on some prior assumptions. While these equations can be used for various dynamic scenarios, such as pandemics, they can also be extended to model the dynamics of two species that are sharing a natural habitat, while competing for a certain local resource, instead of one species just hunting the other. During the last years many different contributions appeared in many fields of applications and in different technical estimation and identification contexts [1], [2], [3].

The general goal of modelling is to find a simple model that fits a data set within a predetermined error bound, while still allowing specific properties to be addressed [4]. In this paper, a least squares method is used to estimate the initial parameter values in the Lotka-Volterra model. This algorithm can be used to estimate the initial value of these parameters to be used in more complex adaptive algorithms as for instance proposed in [5]. The data used is graphics processing units (GPU) shipment worldwide by three key competitors, namely Nvidia, Intel, AMD. As the parameter values of the Lotka-Volterra model describe growth, predation and competition between interacting populations, the model can be applied to competing firms in a market of GPUs or similar. The goal therefore is to quantify the parameter values of a model with minimal error in order to qualitatively solve the problem and fit the raw data as closely as possible. The paper is organized

as follows. Section II proposes a general interpretation of the Lotka-Volterra model. In Section III an application to the market of GPUs is described. Results which include a validation of the method using real data and conclusions close the paper.

II. INTERPRETATION OF THE LOTKA-VOLTERRA MODEL

The Lotka-Volterra equations, also known as the predator-prey equations, are two first-order nonlinear differential equations, often used to describe the dynamics of the biological system in which two species interact, one as a predator and the other as prey. The dynamics of the populations are described as follows:

$$\begin{aligned}\frac{dx(t)}{dt} &= \alpha x(t) - \beta x(t)y(t) \\ \frac{dy(t)}{dt} &= \delta x(t)y(t) - \gamma y(t),\end{aligned}\tag{1}$$

where $x(t)$ represents the number of prey (e.g. fish) and $y(t)$ represents the number of predators (e.g. bears). $\frac{dx(t)}{dt}$ and $\frac{dy(t)}{dt}$ represent the growth rates of the populations while t represents the time variable. Greek letters α , β , δ , γ are constant positive real parameters which describe the interaction of the two groups of populations. While the classic model is usually used in biology, there is a wide range of research with modified models in different areas that resemble the context of predator and prey [4], [6].

III. APPLICATION TO THE MARKET OF GPUS

Differential equations or stochastic differential equations are used for modeling random phenomena in the financial field. Due to the application of these tools in the field of financial economics, a large number of scientists research in this area, see [7], [8], [9]. It is known that the parameters of a stochastic model are always deterministically unknown and this justifies the KF approach mentioned above. The parameter estimation problem for economical models has been studied by many scientists, Yu and Phillips [10] utilized a Gaussian method to estimate the parameters of continuous time short-term interest rate models. Faff and Gray [11] considered the estimation of

short-rate models by using a method of moments. Rossi [12] took into consideration particle filters and maximum likelihood estimation for parameter estimation of Cox-Ingersoll-Ross model. In any market with competition, companies are competing for sales of their proprietary goods and services. An example of a generic Lotka-Volterra model describing the dynamics between three competing PC graphics processing units (GPUs) companies, based on a set of statistical data, is used to illustrate the practical application of the proposed method. The statistical data was retrieved from Statista [13] which includes data from 2nd quarter of 2009 to 3rd quarter of 2021, by vendor (Figure 1).

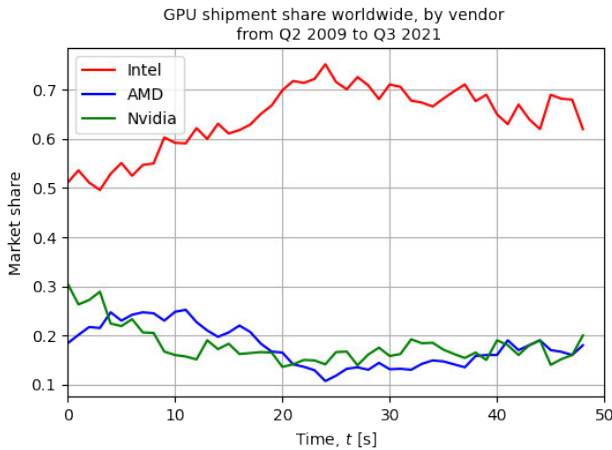


Fig. 1: GPU shipment share worldwide, by vendor, from Q2 2009 to Q3 2021. Source: [13]

From the retrieved data, it can be seen that Intel has the leading position among three firms, while other two firms have a smaller share of the market, therefore are directly competing with each other. This interaction can be modelled using the Lotka-Volterra method and the problem is formulated in terms of a nonlinear dynamical system consisting of three first order differential equations, each containing linear and quadratic terms. The system is linear with regards to the unknown parameters that must be derived from available statistical data. When applying the proposed method to the system, the unknown coefficients are estimated and the closeness of the respective solutions to the statistical data is discussed.

Given a situation where three species x , y and z compete for available resources in a system. In the market of GPUs, three main firms are competing for the attention and resources of similar customers. A Lotka-Volterra model for three competing species is generically represented by:

$$\begin{aligned} \frac{dx(t)}{dt} &= x(t)(a_0 + a_1x(t) + a_2y(t) + a_3z(t)) \\ \frac{dy(t)}{dt} &= y(t)(b_0 + b_1x(t) + b_2y(t) + b_3z(t)) \\ \frac{dz(t)}{dt} &= z(t)(c_0 + c_1x(t) + c_2y(t) + c_3z(t)), \end{aligned} \quad (2)$$

where $x(t)$, $y(t)$, $z(t)$ values are companies' shares in the market and a_i , b_i , c_i ($i = [0, 3]$) are coefficients. To estimate the parameter values of the system of equations (2), the integral method described by [4] was used. As the statistical data is available, it is possible to represent the set of linear equations derived by using the integral method in the matrix form. An example of parameter value estimation for the first firm (Intel) is given below:

$$\underbrace{\begin{bmatrix} d_{1,0} \\ d_{2,1} \\ \vdots \\ d_{n,n-1} \end{bmatrix}}_d = \underbrace{\begin{bmatrix} \bar{x}_{1,0} & \bar{x}_{1,0}^2 & \bar{x}\bar{y}_{1,0} & \bar{x}\bar{z}_{1,0} \\ \bar{x}_{2,1} & \bar{x}_{2,1}^2 & \bar{x}\bar{y}_{2,1} & \bar{x}\bar{z}_{2,1} \\ \vdots & \vdots & \vdots & \vdots \\ \bar{x}_{n,n-1} & \bar{x}_{n,n-1}^2 & \bar{x}\bar{y}_{n,n-1} & \bar{x}\bar{z}_{n,n-1} \end{bmatrix}}_X \underbrace{\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix}}_{\hat{a}}$$

and thus

$$\Rightarrow d = X\hat{a} \quad (3)$$

with

$$d_{j+1,j} = x(t_{j+1}) - x(t_j), j = 0, 1, \dots, (n-1).$$

The matrix X contains the consecutive pair-wise means, namely:

$$\begin{aligned} \bar{x}_{j+1,j} &= (x(t_{j+1}) + x(t_j))/2 \\ \bar{x}_{j+1,j}^2 &= (x^2(t_{j+1}) + x^2(t_j))/2 \\ \bar{x}\bar{y}_{j+1,j} &= (x(t_{j+1})y(t_{j+1}) + x(t_j)y(t_j))/2 \\ \bar{x}\bar{z}_{j+1,j} &= (x(t_{j+1})z(t_{j+1}) + x(t_j)z(t_j))/2. \end{aligned} \quad (4)$$

While the vector a contains the unknown parameters, a least squares method can be used to determine the vector values using the following formula:

$$\hat{a} = (X'X)^{-1}X'd \quad (5)$$

IV. RESULTS

Applying the proposed integral method, the resulting system is given as follows:

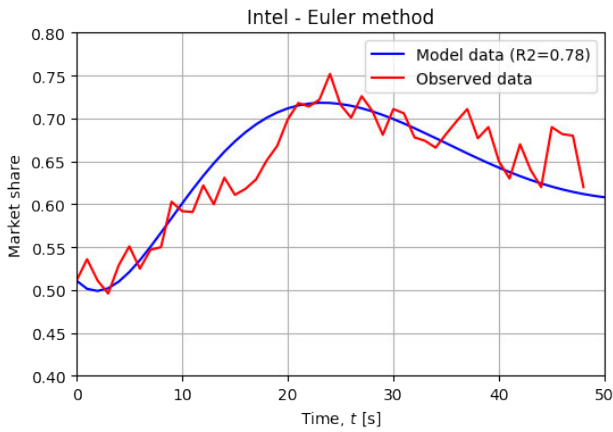
$$\begin{aligned} \frac{dx(t)}{dt} &= x(t)(0.1113 - 0.1122x(t) + 0.1749y(t) - 0.3438z(t)) \\ \frac{dy(t)}{dt} &= y(t)(-0.1297 + 0.0041x(t) - 0.3322y(t) + 0.9385z(t)) \\ \frac{dz(t)}{dt} &= z(t)(0.0804 + 0.0496x(t) - 0.5497y(t) - 0.3117z(t)). \end{aligned} \quad (6)$$

The estimated coefficients show that firms Intel and Nvidia are prey, while the firm AMD is the predator given its negative growth rate of -0.1297 . Interestingly the share of Intel is increased by the actions of the predator firm with a coefficient of $+0.1749$ while Nvidia has a negative influence on Intel's growth rate. This can be seen by a positive coefficient of 0.0496 in the third equation. Looking at the second equation of the predator, it can be seen that the first firm is not acting as the primary prey (0.0041), whereas the third firm is the main source of the predator's share growth, as seen by a

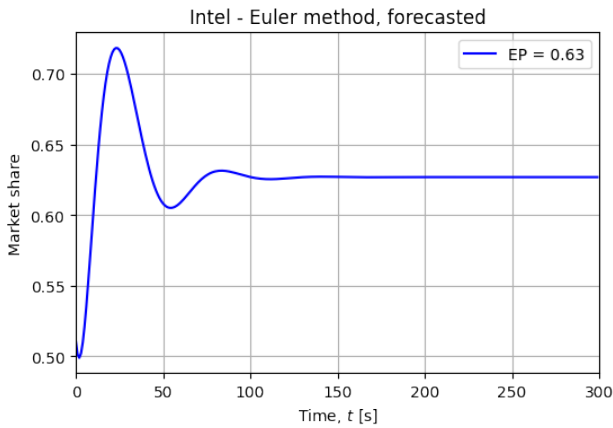
large coefficient of 0.9385. Comparing the second and third equations (AMD vs. Nvidia) it is confirmed by the coefficient of -0.5497 that the growth of AMD comes from the decline in the third firm, Nvidia. This supports the idea that these two firms are direct competitors fighting for the remaining market share as seen on Figure 1.

In order to estimate the equilibrium point for each firm, each equation in the system of equations (6) was set equal to zero. The obvious solution for all equations is when $x(t)$, $y(t)$, $z(t)$ are equal to 0. Another solution was found at points of $x(t) = 0.63$, $y(t) = 0.11$, $z(t) = 0.17$ (see calculation in the code provided).

In order to get the values for model data, the Euler method was used. The following figures (2-4) represent the observed data, the results of modelling and forecasted data for three firms. As seen by the results, the model fits well to historical



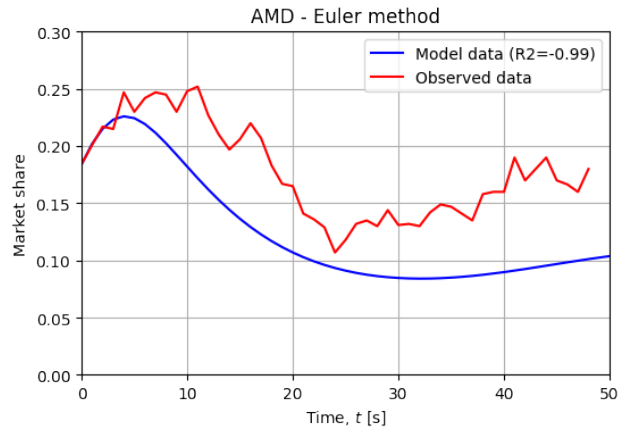
(a) Model and observed data



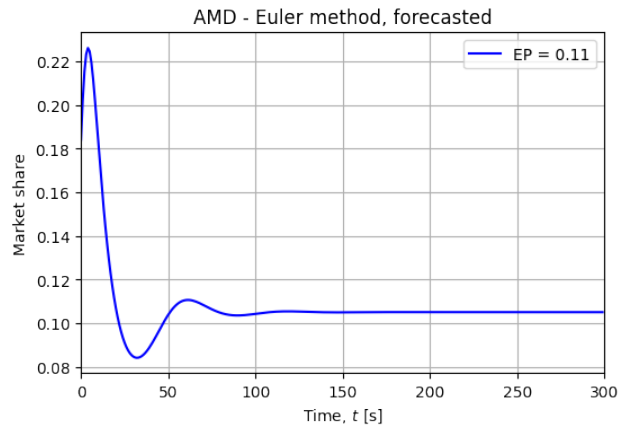
(b) Forecasted data

Fig. 2: The results for Intel

data of Intel market share, with R-squared equal to 0.78. The forecast shows that the stable state is reached at 0.63 that corresponds to the previously retrieved point of equilibrium for Intel. The forecast implies that this firm is likely to maintain its dominance in the GPU market over time.



(a) Model and observed data



(b) Forecasted data

Fig. 3: The results for AMD

As for AMD, the predator firm, it is seen that the model does not fit the observed data, which is confirmed by the R-squared value of -0.99 . When finding the R-squared value, if the sum of squares of the residuals is higher than the sum of the squares of the distances of the points from a horizontal line through the mean of all Y values, then R-squared can be negative. This implies that the "best-fit" line fits worse than a horizontal line drawn through the mean of all Y values, for instance if the regression line does not follow the trend of the observed data [14]. However, as seen on Figure 3a, the modelling data nonetheless displays a similar trend as the observed data but prediction tends to be less accurate than the average value of the data set over time. The forecast implies that AMD would lose half of its market share in the near term and will remain at an equilibrium point of 0.11. However, as the model did not fit well to the observed data, it is impossible to suggest the validity of the forecast. The model for Nvidia shows a good fit with R-squared value equal to 0.78 with an equilibrium point at 0.17. The forecast shows that the firm would lose some of its market share but only by a few percentage points.

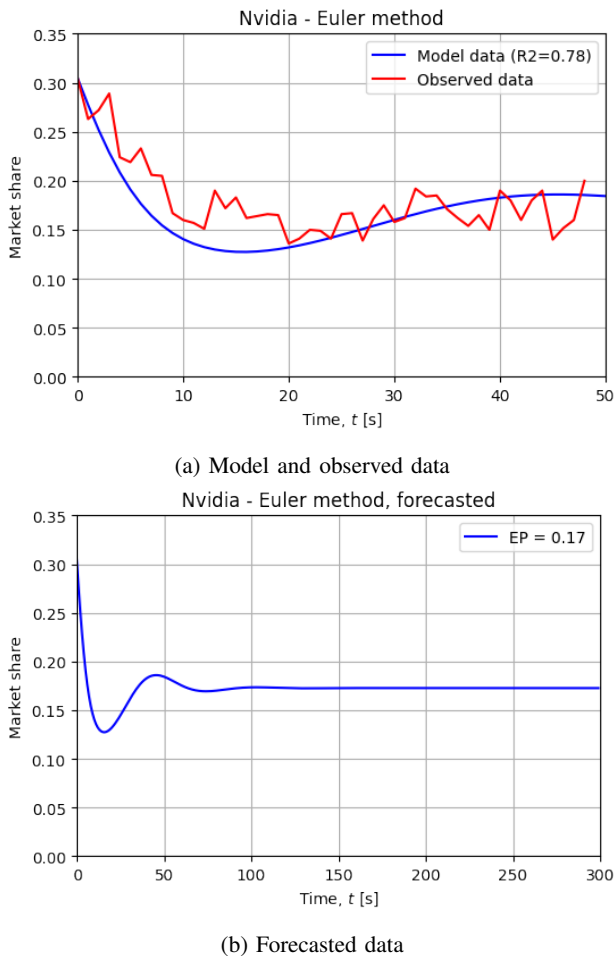


Fig. 4: The results for Nvidia

V. CONCLUSION

In this paper a research in the market of GPUs with three main competing firms was performed using the Lotka-Volterra model. For parameter values estimation the integral method and least squares algorithm was used. The results indicate that AMD is the predator firm while Intel and Nvidia are its prey. However, the R-squared values of the model show decent values for Intel and Nvidia, while not being a good fit for AMD, the predator firm. It is likely that more data is needed to build a better model for the predator firm. Nevertheless, the results indicate that the Lotka-Volterra model can be applied to investigate the competition of more than two firms in a free market of GPUs, but it requires further investigation and experiments.

Acknowledgments:

This work was inspired by the lecture "Modelling and Control of Dynamical Systems using Linear and Nonlinear Differential Equations" held by Paolo Mercorelli within

the scope of the Complementary Studies Programme at Leuphana University of Lüneburg during the winter semester 2021-2022. In this framework, students can explore other disciplinary and methodological approaches from the second semester onwards, focussing on additional aspects in parallel with their subjects and giving them the opportunity to sharpen skills across disciplines.

REFERENCES

- [1] P. Mercorelli, "A Hysteresis Hybrid Extended Kalman Filter as an Observer for Sensorless Valve Control in Camless Internal Combustion Engines," *IEEE Trans on Ind. Appl.*, vol. 48, no. 6, pp. 1940–1949, 2012. [Online]. Available: <https://doi.org/10.1109/TIA.2012.2226193>
- [2] P. Mercorelli, "A Two-Stage Augmented Extended Kalman Filter as an Observer for Sensorless Valve Control in Camless Internal Combustion Engines," *IEEE Trans on Ind. Elects.*, vol. 59, no. 11, pp. 4236–4247, 2012. [Online]. Available: <https://doi.org/10.1109/TIE.2012.2192892>
- [3] K. Benz, C. Rech, and P. Mercorelli, "Sustainable Management of Marine Fish Stocks by Means of Sliding Mode Control," pp. 907–910, 2019. [Online]. Available: <http://dx.doi.org/10.15439/2019F221>
- [4] P. Klopper and J. Greeff, "Lotka-Volterra model parameter estimation using experiential data," *Applied Mathematics and Computation*, vol. 224, pp. 817–825, 2013. [Online]. Available: <https://doi.org/10.1016/j.amc.2013.08.093>
- [5] K. Benz, C. Rech, P. Mercorelli, and O. Sergiyenko, "Two Cascaded and Extended Kalman Filters Combined with Sliding Mode Control for Sustainable Management of Marine Fish Stocks," *Journal of Automation, Mobile Robotics and Intelligent Systems*, pp. 28–35, Jul. 2019. [Online]. Available: <https://doi.org/10.14313/jamris/3-2020/30>
- [6] C. Michalakelis, T. Sphicopoulos, and D. Varoutas, "Modeling Competition in the Telecommunications Market Based on Concepts of Population Biology," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 41, no. 2, pp. 200–210, 2011. [Online]. Available: <https://doi.org/10.1109/tsmcc.2010.2053923>
- [7] J. P. N. Bishwal, *Parameter estimation in stochastic differential equations*, 2008th ed., ser. Lecture Notes in Mathematics. Berlin, Germany: Springer, Oct. 2007.
- [8] X. Zhang, R. D. Brooks, and M. L. King, "A Bayesian approach to bandwidth selection for multivariate kernel regression with an application to state-price density estimation," *Journal of Econometrics*, vol. 153, no. 1, pp. 21–32, Nov. 2009. [Online]. Available: <https://doi.org/10.1016/j.jeconom.2009.04.004>
- [9] W. Xiao, W. Zhang, and W. Xu, "Parameter estimation for fractional Ornstein–Uhlenbeck processes at discrete observation," *Applied Mathematical Modelling*, vol. 35, no. 9, pp. 4196–4207, Sep. 2011. [Online]. Available: <https://doi.org/10.1016/j.apm.2011.02.047>
- [10] J. Yu and P. C. B. Phillips, "A Gaussian approach for continuous time models of the short-term interest rate," *The Econometrics Journal*, vol. 4, no. 2, pp. 210–224, Dec. 2001. [Online]. Available: <https://doi.org/10.1111/1368-423x.00063>
- [11] R. Faff and P. Gray, "On the estimation and comparison of short-rate models using the generalised method of moments," *Journal of Banking & Finance*, vol. 30, no. 11, pp. 3131–3146, Nov. 2006. [Online]. Available: <https://doi.org/10.1016/j.jbankfin.2005.09.016>
- [12] G. D. Rossi, "Maximum Likelihood Estimation of the Cox–Ingersoll–Ross Model Using Particle Filters," *Computational Economics*, vol. 36, no. 1, pp. 1–16, Mar. 2010. [Online]. Available: <https://doi.org/10.1007/s10614-010-9208-0>
- [13] Statista, "PC GPU shipment share worldwide Q2 2009 - Q3 2021, by vendor," 2021, data retrieved from Statista, <https://www.statista.com/statistics/754557/worldwide-gpu-shipments-market-share-by-vendor/>.
- [14] D. Chicco, M. J. Warrens, and G. Jurman, "The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation," vol. 7, p. e623. [Online]. Available: <https://doi.org/10.7717/peerj-cs.623>

Tag and correct: high precision post-editing approach to correction of speech recognition errors

Tomasz Ziętkiewicz

Adam Mickiewicz University in Poznań

ul. Uniwersytetu Poznańskiego 4, 61-614 Poznań, Poland

Samsung R&D Institute Poland

Plac Europejski 1, 00-844 Warsaw, Poland

Email: t.zietkewic@samsung.com, tomasz.zietkewicz@amu.edu.pl

Abstract—This paper presents a new approach to the problem of correcting speech recognition errors by means of post-editing. It consists of using a neural sequence tagger that learns how to correct an ASR (Automatic Speech Recognition) hypothesis word by word and a corrector module that applies corrections returned by the tagger. The proposed solution is applicable to any ASR system, regardless of its architecture, and provides high-precision control over errors being corrected. This is especially crucial in production environments, where avoiding the introduction of new mistakes by the error correction model may be more important than the net gain in overall results. The results show that the performance of the proposed error correction models is comparable with previous approaches, while requiring much smaller resources to train, which makes it suitable for industrial applications, where both inference latency and training times are critical factors that limit the use of other techniques.

Index Terms—speech recognition, error correction, post-processing, post-editing, natural language processing

I. INTRODUCTION

AUTOMATIC Speech Recognition (ASR) models have been developed for more than 70 years. During this time, they evolved from machines that could recognize single digits spoken by one person, created for demonstration purposes without production use back in the 1950s [1], to omnipresent voice assistants and speech transcription engines used everyday by millions of people around the world. Although speech recognition technology has reached maturity and is production ready, it is still not perfect. Professional human transcribers do not reach 100% transcription accuracy, and although recent deep neural network-powered speech recognition systems are reported to slightly outperform humans [2], they still make mistakes. Furthermore, the near-perfect results of the Word Error Rate (WER) below 2% are often reported on popular benchmark datasets such as the test subset of the LibriSpeech corpus [3]. When evaluated on corpora from different domains or recorded under different conditions, the results are far from perfect and require adaptation [4]. In real-world production settings, input to a speech recognition system may change frequently, caused by changes in topics that are interesting to users. Changes in the real world, such as the Covid-19 pandemic, influence the vocabulary used by speakers and may lead to out-of-vocabulary errors. The adaptation process of ASR models takes considerable time and resources. In some

settings, direct adaptation of the model is impossible, for example, when using a cloud-based speech recognition service. One way to address the imperfections of speech recognition models mentioned above is to improve their results in a post-editor, a module operating on a textual output of a speech recognition model. Typically, in production environments, the post-editor provides means of correcting ASR errors manually, for example, with hand-written regular expressions. The process of creating and maintaining such corrections is laborious and requires a lot of experience [5]. Therefore, the post-editor can include an error correction model which learns how to correct errors of a particular ASR system. This approach can be applied regardless of an ASR model architecture, also for systems where direct modification of the model is impossible. It requires only text data to train, and its training requires considerably less resources and time than performing the adaptation of a speech recognition model. This paper presents such an error correction model. In Section II we present an overview of previous work on the subject. Section III describes the data used for training and evaluation. Details of the proposed approach are presented in Section IV. Results and conclusions can be found in Sections V and VI.

II. RELATED WORK

For a review of ASR error detection and correction systems together with a description of ASR evaluation metrics see [6]. Cucu et al. [7] propose error correction using SMT (Statistical Machine Translation) model trained on a relatively small parallel corpus of 2000 ASR transcripts and their manually corrected versions. At an evaluation time, the model is used to “translate” ASR hypothesis into corrected form. The system achieves 10.5% relative WER improvement by reducing WER of the baseline ASR system from 11.4 to 10.2. A similar approach, but using a neural LSTM sequence-to-sequence model and trained on a much larger dataset (40M utterances), is presented in [8]. To produce ASR hypotheses, the authors use a speech corpus generated from plain text data with a text-to-speech (TTS) system. In addition to the spelling correction model, authors experiment with improving the results of an end-to-end ASR system by incorporating an external language model and a combination of the two approaches. The proposed system achieves satisfactory results (19% rela-

tive WER improvement and 29% relative WER improvement with additional LM re-scoring, with baseline ASR WER of 6.03) but requires a large speech corpus or high-quality TTS system to generate such corpus from a plain text. One of the recent works [9] presents an error correction model for Mandarin. The authors stress the importance of a low latency of ASR error correction model in production environments and propose a non-autoregressive transformer model, faster than its autoregressive counterparts. The model is modeled on a large, artificially created parallel corpus of correct-incorrect sentence pairs, generated by randomly deleting, inserting, and replacing words in a text corpus. Real ASR corrections dataset is used to fine-tune the model to a specific ASR system. Relative WER reduction reported by authors on a publicly available testset is 13.87, which is slightly worse than autoregressive model (15.53) while introducing a latency that is over 6 times lower (21ms).

A similar approach to the one presented in this work is proposed in [10], but serves different tasks (grammatical error correction), operates on a poorer set of edit operations and uses different tagging models.

III. DATA

We performed experiments for 3 European languages: Spanish, French, and German. The presented error correction models were trained and evaluated on pairs of ASR hypotheses and corresponding reference sentences. To create a corpus of such pairs, recordings from speech corpora for each language were processed using a corresponding model of an end-to-end speech recognition system. Reference transcriptions from speech corpora were paired with their corresponding hypotheses, creating parallel corpora of corrections for each language. To discard any differences caused by different normalization of transcriptions in speech corpora and in ASR output, we additionally performed automatic normalization of both reference and hypotheses sentences by lowercasing, removing punctuation characters and inverse-normalizing numbers. The data preparation pipeline is presented in Figure 1. For an example of a freely available corpus of ASR corrections for Polish prepared with the same pipeline, see [11].

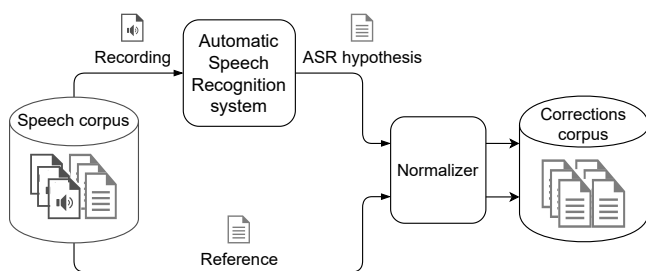


Fig. 1. Data preparation pipeline

Speech corpora, used to create corrections corpora, contain utterances used for virtual assistant development. They include commands and questions targeted at the assistant. Some of them are recorded in a studio for development purposes, and

TABLE I
DATASETS STATISTICS

	de-DE	es-ES	fr-FR
Sentences	12242	16905	7180
Tokens	37955	55567	28004

some originate from usage data, after performing anonymization.

Data statistics are shown in Table I. Data sets were randomly divided into training, development, and test subsets in a proportion 8:1:1.

For a description of tagger training data preparation process, see section IV-A.

IV. METHOD

The proposed error correction method is designed to be precise, easily controllable, and data efficient. In contrast to methods inspired by machine translation, such as [8], our model does not have to model and reproduce all tokens of the output sequence. Instead, it learns only which tokens to modify to correct the sentence. To make it precisely controllable, the error correction mechanism works in two steps. First, a sequence-tagging model assigns a tag to each token in the input sentence. The tag indicates whether the token is correctly recognized or requires correction. In the latter case, the tag specifies an edit operation that is needed to transform the incorrect sentence into a correct one. In the second step, the assigned tags are used to perform edit operations that correct ASR errors. This approach is especially suitable for production settings because it allows one to precisely control which edit operations to include in the model and which edit operations to perform on inference time. The control can be based on scores returned by the tagger (for example, by setting a global scorer threshold) or on rules excluding certain operations in certain contexts from being performed.

A. Tagging data preparation

To train a sequence tagging model, a corpus of ASR hypotheses with assigned edit operation tags is needed. We create it from the parallel corrections corpus described in Section III. First, hypothesis and reference sentences are compared and aligned using Ratcliff-Obershelp algorithm [12] implemented in difflib library [13]. As a result, we get a list of operation codes describing how to turn corresponding parts of the hypothesis into reference. There are four operation codes: "replace", "delete", "insert" and "equal". For parts of a sentence which are not equal, we use a set of conditional rules involving recursively invoking the difflib alignment to tag each token with one of the edit operations. Examples of tags and corresponding edit operations are presented in table II. For an illustrated example of tags generation process, see Figure 2.

The set of available edit operation classes can be adjusted to the needs of a specific speech recognition system and a natural language. Ideally, the set should cover all errors and be as small as possible. If the set is too big, edit operations become

too sparse for the tagger to learn them effectively. Therefore, we chose to prioritize use of most expressive edit operations, like „append_s”, which for example in English could cover most of the errors associated with a singular noun in place of a plural and missing "s" in third-person singular verbs. The same errors could be covered by more precise rules correcting only particular words, e.g. "replace_with_cats" (edit operation assigned to the word "cat"). By using more general operations, the model can generalize to unseen examples of errors. This is where our approach is different from [10] which uses only two main types of operations: KEEP ("None" in our approach), DELETE ("del" in our approach) and can insert any phrase or word from vocabulary V before the current token. Such an approach cannot cover multiple errors with one tag.

Despite using edit operations that cover multiple errors, the edit operations dataset is still sparse. About 15% of edit operations are supported only with one example. To make training more efficient and the model less overfitted to singular examples, we use a cut-off of 150 most frequent edit operations, also filtering-out all which are found only once in the dataset. To differentiate between tokens which are correct and those which errors are not frequent enough, we replace all the filtered edit operations with a special edit operation called "unsupported". This operation tag is present in the training set and the model learns to tag some tokens with it, but when correcting sentences on the inference time, there is no edit operation performed on them. Sequence of edit operations assigned to adjacent tokens can be interdependent - performing only some of them may deteriorate the results. Therefore, when one of the tokens is tagged with "unsupported", all surrounding, non-empty tags are replaced with the "unsupported" tag.

B. Tagging models

We train and evaluate two tagging model types: BERT token classification model and contextual string embeddings model [14], referred later as "Flair". BERT tagger model uses locale variations of BERT transformer: Gbert for German [15], Camembert for French [16] and Beto for Spanish [17]. A single linear layer is added at the output, and the whole network is fine-tuned for the tagging task. Training was performed for 6 epochs, extending the training time did not improve results. The "Flair" model contains Bert models mentioned before, extended with contextual string embeddings [18], LSTM [19] and CRF [20] layers. Training was carried out for a maximum of 100 epochs.

V. RESULTS

Both model types were evaluated using hold-out test sets by calculating WRR (Word Recognition Rate) of original ASR hypotheses and their corrected versions. Table III presents averaged results. As can be seen on an example of German dataset compared with other two - the worse the original ASR results are, the easier for the correction model to achieve higher absolute gains. Therefore, to compare correction models trained on datasets with different original results, we

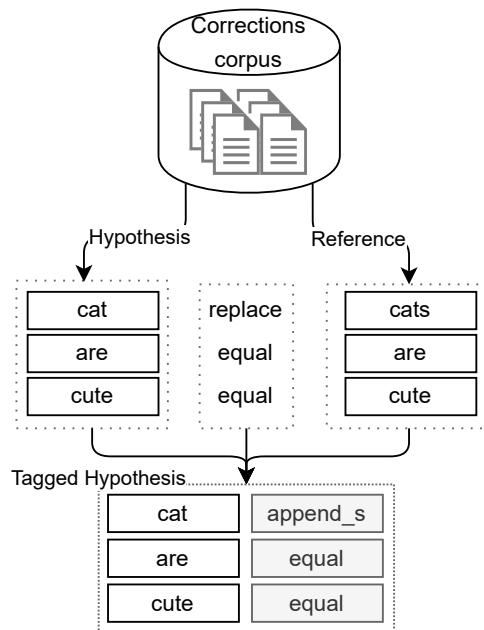


Fig. 2. Tags generation example

calculate a relative WER (Word Error Rate) reduction metric, which is calculated as

$$\frac{WER_{ASR} - WER_{Corrected}}{WER_{ASR}} \quad (1)$$

where WER_{ASR} is WER of ASR system and $WER_{Corrected}$ is WER after applying the correction model. Relative improvements of our models vary between 21% and 24.7%, making them comparable with state-of-the-art results reported in [8], while using much smaller training datasets, without the need of generating synthetic data with TTS engine. The Flair tagger offers slightly better results (except for German, where results are equal).

Table IV presents the time required for training the models and the average times needed to correct a single sentence from the test set. Both training and inference were performed using a machine with a single Tesla P40 GPU.

VI. CONCLUSIONS

We presented a new approach to ASR errors correction problem. As demonstrated using three independent datasets, correction models trained using this approach are effective even for relatively small training datasets. The method allows to precisely control which errors should be included in the model and which of the included ones should be corrected at the inference time. The evaluations performed on the models show that they can significantly improve the ASR results by reducing the WER by more than 20%. All of the models presented offer very good inference latency, making them suitable for use with streaming ASR systems.

TABLE II
EXAMPLES OF EDIT OPERATIONS

name	description	example
del	deletes a token	"a" → ""
append_s	appends given suffix to the token	"cat" → "cats"
add_prefix_	prepends given prefix to the token	"owl" → "howl"
remove_suffix_1	removes 1 character from the end of the token	"cats" → "cat"
remove_prefix_1	removes 1 character from the beginning of the token	"howl" → "owl"
join	joins token with previous one	"book store" → "bookstore"
join_	joins token with previous one using given separator	"long term" → "long-term"
replace_	replaces token with given string	"cat" → "hat"

TABLE III
RESULTS OF ERROR CORRECTION MODELS

		de-DE	es-ES	fr-FR
BERT	base WRR	78.07	90.70	93.51%
	corrected WRR	83.48	92.65	94.97%
	WRR gain	5.41	1.95	1.46
	Rel. WER reduction	24.67%	20.97%	22.50%
Flair	corrected WRR	83.40	92.86	95.04
	WRR gain	5.33	2.16	1.53
	Rel. WER reduction	24.30%	23.23%	23.57%

TABLE IV
TRAINING (IN MINUTES) AND INFERENCE (IN MILLISECONDS) TIMES.

		de-DE	es-ES	fr-FR
BERT	training time	28	40	10
	inference latency	14	14	13
Flair	training time	180	154	67
	inference latency	28	29	18

The presented method is well suited for industrial applications where the ability to precisely control how the error correction model works, as well as small latency, are crucial.

REFERENCES

- [1] K. H. Davis, R. Biddulph, and S. Balashek, "Automatic recognition of spoken digits," *The Journal of the Acoustical Society of America*, vol. 24, no. 6, pp. 637–642, 1952. [Online]. Available: <https://doi.org/10.1121/1.1906946>
- [2] W. Xiong, J. Droppo, X. Huang, F. Seide, M. L. Seltzer, A. Stolcke, D. Yu, and G. Zweig, "Achieving human parity in conversational speech recognition," *ArXiv*, vol. abs/1610.05256, 2016.
- [3] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An asr corpus based on public domain audio books," *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5206–5210, 2015.
- [4] A. Narayanan, A. Misra, K. C. Sim, G. Pundak, A. Tripathi, M. Elfeky, P. Haghani, T. Strohmman, and M. Bacchiani, "Toward domain-invariant speech recognition via large scale training," in *2018 IEEE Spoken Language Technology Workshop (SLT)*, 2018, pp. 441–447.
- [5] A. Jeziorski, F. Sawicki, O. Solop, M. Junczyk, M. Sikora, and T. Zietkiewicz, "Industrial asr troubleshooting tool," in *Proceedings of the LREC2020 Industry Track*. Marseille, France: European Language Resources Association (ELRA), May 2020, pp. 10–14.
- [6] R. Errattahi, A. El Hannani, and H. Ouahmane, "Automatic speech recognition errors detection and correction: A review," *Procedia Computer Science*, vol. 128, pp. 32–37, 01 2018.
- [7] H. Cucu, A. Buzo, L. Besacier, and C. Burileanu, "Statistical Error Correction Methods for Domain-Specific ASR Systems," in *Statistical Language and Speech Processing*, A.-H. Dediú, C. Martín-Vide, R. Mitkov, and B. Truthe, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 83–92.
- [8] J. Guo, T. N. Sainath, and R. J. Weiss, "A spelling correction model for end-to-end speech recognition," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 5651–5655. [Online]. Available: <https://arxiv.org/pdf/1902.07178>
- [9] Y. Leng, X. Tan, L. Zhu, J. Xu, R. Luo, L. Liu, T. Qin, X.-Y. Li, E. Lin, and T.-Y. Liu, "Fastcorrect: Fast error correction with edit alignment for automatic speech recognition," 2021.
- [10] E. Malmi, S. Krause, S. Rothe, D. Mirylenka, and A. Severyn, "Encode, tag, realize: High-precision text editing," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 5054–5065. [Online]. Available: <https://aclanthology.org/D19-1510>
- [11] M. Kubis, Z. Vetulani, M. Wypych, and T. Zietkiewicz, "Open challenge for correcting errors of speech recognition systems," in *Proceedings of the 9th Language and Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics*, Z. Vetulani and P. Paroubek, Eds. Poznań, Poland: Wydawnictwo Nauka i Innowacje, 2019, pp. 219–223. [Online]. Available: <https://arxiv.org/abs/2001.03041><https://gonito.net/gitlist/asr-corrections.git/>
- [12] D. E. M. John W. Ratcliff, "Pattern matching: The gestalt approach," p. 46, 7 1988. [Online]. Available: <https://www.drdoobs.com/database/pattern-matching-the-gestalt-approach/184407970>
- [13] "difflib — helpers for computing deltas," <https://docs.python.org/3/library/difflib.html>, accessed: 2022-03-20.
- [14] A. Akbik, T. Bergmann, D. Blythe, K. Rasul, S. Schweter, and R. Vollgraf, "FLAIR: An easy-to-use framework for state-of-the-art NLP," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*. Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 54–59. [Online]. Available: <https://www.aclweb.org/anthology/N19-4010>
- [15] B. Chan, S. Schweter, and T. Möller, "German's next language model," in *Proceedings of the 28th International Conference on Computational Linguistics*. Barcelona, Spain (Online): International Committee on Computational Linguistics, Dec. 2020, pp. 6788–6796. [Online]. Available: <https://aclanthology.org/2020.coling-main.598>
- [16] L. Martin, B. Müller, P. J. Ortiz Suárez, Y. Dupont, L. Romary, É. de la Clergerie, D. Seddah, and B. Sagot, "CamemBERT: a tasty French language model," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, Jul. 2020, pp. 7203–7219. [Online]. Available: <https://www.aclweb.org/anthology/2020.acl-main.645>
- [17] J. Cañete, G. Chaperon, R. Fuentes, J.-H. Ho, H. Kang, and J. Pérez, "Spanish pre-trained bert model and evaluation data," in *PMLADC at ICLR 2020*, 2020.
- [18] A. Akbik, D. Blythe, and R. Vollgraf, "Contextual string embeddings for sequence labeling," in *Proceedings of the 27th International Conference on Computational Linguistics*. Santa Fe, New Mexico, USA: Association for Computational Linguistics, Aug. 2018, pp. 1638–1649. [Online]. Available: <https://aclanthology.org/C18-1139>
- [19] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, pp. 1735–80, 12 1997.
- [20] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of the Eighteenth International Conference on Machine Learning*, ser. ICML '01. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001, pp. 282–289. [Online]. Available: <http://dl.acm.org/citation.cfm?id=645530.655813>

Author Index

- A**
Abid, Amal 685
Abramowicz, Witold 705
Adrian, Marek 501
Alt, Rainer 773
Amin, Md Nur 129
Angelova, Denitsa 329
Antkiewicz, Michał 291
Artiemjew, Piotr 149
Atanassov, Krassimir 1
- B**
Bączkiewicz, Aleksandra 769, 799
Bal, Seval 883
Bambul-Mazurek, Elżbieta 751
Barros, Rodolfo Miranda de 721
Barros, Vanessa Tavares de Oliveira 721
Bechikh, Slim 253
Bicevska, Zane 763
Bicevskis, Janis 763
Bielecki, Włodzimierz 475
Bieniek-Majka, Maryla 741
Bierlaire, Michel 301
Binnewitt, Johanna 323
Blum, Christian 363
Bolle, Sebastien 641
Borowicz, Adam 305
Bosse, Stefan 601
Buczyński, Hubert 605
Bureva, Veselina 1
Bylina, Beata 479
Bylina, Jarosław 479
- C**
Cabaj, Krzysztof 605
Campagner, Andrea 243
Caputa, Jakub 169
Carlson, Jan 907
Cassia, Antonio 313
Cen, Ling 431, 447
Cheikhrouhou, Saoussen 685
Ciucci, Davide 243
Cornelis, Chris 7, 269
Cyganeł, Bogusław 35
Czejdo, Danny 913
- D**
Dąbrowska-Boruch, Agnieszka 169
Dagdia, Zaine Chelly 253
Daszczuk, Wiktor 913
David, Istvan 849
Descours, Danuta 809
Didkivska, Svitlana 695
- Diego, Isaac Martín De 93
Dietz, Eric 855
Dimov, Ivan 329
Dörpinghaus, Jens 43, 323
Douet, Marc 641
Do, Van Khanh 181
Duch, Włodzisław 741
Du, Dan 451
Dutka, Łukasz 563
Dutta, Soma 751
Dymek, Dariusz 695
Dymora, Paweł 583
- E**
Enoiu, Eduard 907
Erdogan, Tugba Gurgun 883
- F**
Faber, Jennifer 43
Fadda, Edoardo 301
Fareh, Messaouda 733
Ferdowski, Arman 569
Fernandez-Isabel, Alberto 93
Fialko, Sergiy 457
Fidanova, Stefka 329
Filipowska, Agata 227
Fister, Jr., Iztok 109
Frączek, Rafał 169
Furtak, Janusz 611
- G**
Gakh, Dmitriy 701
Ganzha, Maria 329
Gawrysiak, Piotr 141
Geldenuys, Morgan 553
Gjorgovski, Pavel 437
Goczyła, Krzysztof 861
Gök, Mehmet Şahin 883
Gonce, Renaud 635
Górecki, Tomasz 227, 421
Gotówko, Łukasz 593
Grabowski, Mariusz 695
Graliński, Filip 73, 121
Greco, Gianluigi 501
Grodzki, Michał 217, 403
Grzegorowski, Marek 217, 403
Grzeszczyk, Jakub 169
Gustavsson, Henrik 907

H abarta, Filip	489	Kotulski, Zbigniew	673
Hamel, Oussama	733	Kozak, Jan	715
Hansch, Gerhard	653	Kozielski, Michal	89
Hanslo, Ridwaan	53	Kozłowski, Marek	425
Hasimi, Lumbardha	213	Krishna, Aneesh	873
Hengeveld, Simon	175	Kubina, Anna	61
Hengeveld, Simon B.	333	Kulawiak, Marcin	163
Henzel, Joanna	61	Kumar, Lov	873, 895
Hirata, Kouichi	351	Kwieciński, Robert	227
Hrach, Christian	773		
Hristoskova, Anna	635	Ł amasz, Bartosz	357
Hristozov, Anton	855	Lamecki, Andrzej	535
Hryniewicz, Olgierd	931	Lancho, Carmen	93
Hubacz, Marcin	593	Laszczyk, Maciej	291
Hübenthal, Tobias	43	Lepak, Łukasz	133
		Lewandowska, Aleksandra	417
I mran, Abdullah Al	129	Lewoniewski, Włodzimierz	705
		Ligeza, Antoni	817
J abali, Ola	313	Lippi, Marco	15
Jach, Tomasz	715	Liu, Ming	431, 447
Jakobs, Christine	505, 653	Liu, Yuling	451
Jamiołkowski, Antoni	399	Łukasik, Daria	169
Jamro, Ernest	169	Lu, Zhigang	451
Janicki, Ryszard	247	Łyczko, Kamil	583
Janusz, Andrzej	393, 399		
Jarosz, Kamil	563	M aciąg, Piotr S.	79
Jarosz, Michał	617, 627	Mahmoud, Mahmoud	247
Jelonek, Dorota	827	Majima, Seiyo	235
Jerbi, Manel	253	Mala, Ivana	489
Jezusek, Piotr	67	Malucelli, Federico	313
Jmaiel, Mohamed	685	Manerba, Daniele	301
		Marchi, Felipe	337
K aczmarek, Karol	73	Marciniak, Tomasz	117
Kaczmarek-Majer, Katarzyna	931	Marek, Luboš	489
Kaczmarek, Krzysztof	535	Mariani, Stefano	15
Kadry, Seifedine	201	Markov, Konstantin	235
Kallel, Slim	685	Martinelli, Matteo	15
Kamińska, Olga	931	Marzal, Eliseo	347
Kanciak, Krzysztof	627	Masson, Constantin	849
Kannout, Eyad	217, 403	Maszczyk, Cezary	89
Kao, Odej	553	Matelski, Sławomir	663
Karakatič, Sašo	109	Matson, Eric	855
Karpus, Aleksandra	67	Mavrov, Deyan	383, 387
Karwatowski, Michał	169	Mazurek, Mirosław	583
Kasprzak, Włodzimierz	181	Mazurek, Szymon	169
Kassjański, Michał	163	Mercorelli, Paolo	935
Katra, Szymon	913	Michalak, Jarosław	577
Kaźmierczak, Stanisław	413	Mieszkowicz-Rolka, Alicja	263
Kitowski, Jacek	563	Misra, Sanjay	895
Kizielewicz, Bartłomiej	783, 789	Miyazaki, Tomoya	351
Klein, Sarah	635	Młyński, Marcin	149
Kluza, Krzysztof	817	Moguerza, Javier M.	93
Koniarski, Konrad	531	Moreno, Raúl	93
Koryciak, Sebastian	169	Moualla, Ghada	641
Korzeniowski, Łukasz	861	Mrela, Aleksandra	741

Mucherino, Antonio	175, 333
Mula, Venkata Krishna Chandra	873
Munna, Mahmud Hasan	129
Murešan, Horea-Bogdan	103
Murthy, Lalita Bhanu	873, 895
Myśliński, Andrzej	531
Myszkowski, Paweł	291

N akayama, Minoru	745
Naumann, Billy	505
Nguyen, Hung Son	279
Niewolski, Wojciech	673
Nitarska, Natalia	817
Normatov, Dzhakhongir	935
Nowacka, Anna	827
Nowak, Tomasz	191, 673
Nowak, Wioletta	745

O dītis, Ivo	763
Okulewicz, Michał	399
Opiola, Łukasz	563
Osinksa, Veslava	741
Ozkan, Necmettin	883

P acud, Radosław	715
Paliwoda-Pękosz, Grażyna	695
Palkowski, Marek	475
Papaj, Tomasz	809
Paradowski, Bartosz	789
Parfieniuk, Marek	545
Parpinelli, Rafael Stubs	337
Pascoal, Marta	313
Pavlič, Sašo	109
Pawłowicz, Bartosz	593
Pfister, Ben	553
Piekarz, Monika	479
Pietroń, Marcin	169
Pioroński, Sławomir	421
Pisarczyk, Paweł	605
Piwowarski, Paweł	181
Plachetka, Tomas	515
Podbucki, Kacper	117
Podlodowski, Łukasz	425
Pokrywka, Jakub	73, 121
Poliwoda, Maciej	475
Poniszewska-Marañda, Aneta	213
Porter-Sobieraj, Joanna	535
Probierz, Barbara	715
Przewoźny, Tomasz	163
Puka, Radosław	357
Puławski, Łukasz	521

R ath, Annanda	635
Rifat, Md Rifatul Islam	129
Rogers, Marcus	855
Rolka, Leszek	263
Russek, Paweł	169
Ruta, Dymitr	431, 447
Rutten, Eric	641
Rybiński, Henryk	79
Rybiński, Kamil	919
Rypeś, Grzegorz	133

S ackmann, Stefan	773
Sadolewski, Jan	587
Said, Lamjed Ben	253
Saľabun, Wojciech	783, 789
Salach, Mateusz	593
Sanin, Cesar	841
Sartori, Camilo Chacón	363
Sasak-Okoń, Anna	467
Scheinert, Dominik	553
Schmidt, Karsten	653
Sebastia, Laura	347
Sepczuk, Mariusz	673
Sharma, Raghav	837
Sharma, Raksha	837
Shekhovtsov, Andrii	783
Sikora, Marek	61, 89
Singh, Deependra	837
Sitek, Wojciech	79
Skalna, Iwona	357
Skoczylas, Mariusz	593
Skonieczny, Łukasz	79
Skowron, Andrzej	23
Skrzypczyński, Piotr	191
Ślęzak, Dominik	23, 393
Słota, Renata G.	563
Śluzek, Andrzej	205
Smialek, Michał	919
Śmiech, Anna	169
Sokolov, Oleksandr	741
Sosnowski, Łukasz	751
Sosnowski, Witold	141
Spulis, Viesturs	763
Stein, Neta	373
Štěpánek, Lubomír	489
Sterling-Zuluaga, Karina	841
Švaňa, Miloš	157
Syriani, Eugene	849
Szczerbicki, Edward	841
Szymusik, Iwona	751

T amir, Tami	373
Teresa, Marina Cuesta Santa	93
Terra, Marcus Vinicius Alencar	721
Thamsen, Lauritz	553

Theerens, Adnan	269	Werner, Matthias	505, 653
Tiotsop, Lohic Fotio	301	Wiatr, Kazimierz	169
Trajanoska, Milena	437	Więckowski, Jakub	789
Traneva, Velichka	383, 387	Wielgosz, Maciej	169
Tranov, Stoyan	383, 387	Wiśniewski, Piotr	817
Trybus, Bartosz	587, 593	Wróbel, Łukasz	61
Turek, Tomasz	715	Wróblewska, Anna	141
U berg, Lewi	201	Wrona, Konrad	617, 627
V akaliuk, Tetiana Anatoliivna	695	X iao, Haitao	451
Vijayvargiya, Sanidhya	895	Z ambonelli, Franco	15
Vo, Bich Khue	279	Zaremba, Mateusz	817
Vu, Quang Hieu	431, 447	Zarowska, Anna	745
W ątróbski, Jarosław	799	Zdravevski, Eftim	437
Wawrowski, Łukasz	61	Zieliński, Zbigniew	617
Wawrzyński, Paweł	133	Ziomba, Ewa	809
Węcel, Krzysztof	705	Ziętkiewicz, Tomasz	939
Weil, Vera	323	Żuławińska, Joanna	751
Weiss, Aleksandra	149	Zyguła, Aleksandra	751