An Inference Strategy for Knowledge Units

Michal Peták[1], Milan Houška[1]

[1]Department of Systems Engineering, Faculty of Economics and Management,
Czech University of Life Sciences Prague, Kamýcká 129, 165 21 Prague 6 - Suchdol
Czech Republic, e-mail: petak@pef.czu.cz, houska@pef.czu.cz

Abstract. Inference algorithm is an integral part of any expert system based on and working with procedural knowledge such as production rules. In general, also the production rules containing the compound statements in an antecedent of a consequent part is well known and supported with knowledge or expert systems. On the other hand, if a production rule is of an advanced internal structure (is transformed into a knowledge unit), standard inference algorithms (e.g. modus ponens or modus tollens) could provide insufficient results. The aim of this paper is to suggest a modification of the inference algorithms to be appropriate for working with knowledge units. The formal notification of the algorithm is accompanied with a practical example in the domain of Enterprise Resource Planning (ERP) system used for the support of a particular business process.

Keywords: Knowledge unit, Inference mechanisms, Expert system, Knowledge representation.

1 Introduction

Inference engines (methods, strategies, procedures) are present in expert systems used in various problem domains. (Moreno and Espejo, 2015) present a performance evaluation of three different inference engines (rule based reasoning, fuzzy based reasoning and Bayesian based reasoning) for failure mode identification in shafts. They compare different types of inference mechanisms to improve the expert system and conclude that, under their conditions, the best inference mechanisms are Bayesian and Fuzzy rules inference.

(Venturelli et al., 2017) propose a method to evaluate the efficiency of Corporate Social Responsibility (CSR). The outcome of the application is a system designed to measure the CSR identity of a company. The algorithm of the Fuzzy Expert System aggregates multicriteria evaluations of a problem. The assessments of behavior and the resulting decisions are represented in blocks of rules, drawn up by an inference engine in fuzzy logic. The Fuzzy Expert System unites the ability of an expert system to simulate the decision-making process with the uncertainty typical of human reasoning, present in fuzzy logic. The aim is create a model using Fuzzy Expert System approach, which serves to combine the intuition and the experience of the experts who supply a knowledge base with the formal rigor of a logic system.

Proceedings of the 8th International Conference on Information and Communication Technologies in Agriculture, Food and Environment (HAICTA 2017), Chania, Greece, 21-24 September, 2017.

The expert system by (Chen and Pollino, 2012) employed the Bayesian Inferential Network to map a suitable biotope for particular animal species (Juvenile Astacopsis gouldi – giant freshwater crayfish of Tasmania). The results of the inference are visualized with a Geographical Information Systems (GIS).

(Fakhrahmad et al., 2015) use the expert system to reduce ambiguity in automated translation. This expert systemcombinestwo techniques: Forward chaining and Data mining. Forward chaining Word Sense disambiguation (FchWSD) is very efficient because it can disambiguate other ambiguous words existing in the context in addition to the target ambiguous word in just one pass through the knowledge base. The performance of the proposed system in terms of Recall and Precision was encouraging compared to its counterparts.

Technical systems and their modernization or aging are being analyzed with mathematical models Krejci (2013). Regardless, another expert system on microstructural characterization of dual-phase steels was developed by (Ghanei et al., 2015). Thisexpert systembased on an adaptive neuro- fuzzy inference system (ANFIS) helps the human users with evaluation of material state: Evaluation of non-destructive control.

Expert systems are being used to simulate human decision-making process in complex decision situations (Venturelli et al., 2017). For this purpose, they use various kinds of knowledge representations. Mařík et al. (2004) distinguish the procedural and declarative knowledge as follows: procedural knowledge means "knowing how", declarative knowledge means "knowing what".

Knowledge unit is an extended procedural production rule with fixed internal structure (Domeová et al., 2008, see Materials and Methods for its definition). In this paper we propose an advanced inference algorithm suitable for a specific kind of production rules – knowledge units. We show how to use the internal structure of the knowledge unit to determine relationship among individual parts of the precedingknowledge unit and the successing knowledge unit within the inference process. The inference process is also demonstrated on a practical example – particular process within the working with an ERP system.

2 Materials and Methods

2.1 Production Rules and Their Inference

Inference strategy, i.e. the strategy how the system operates the responses to the questions, is the basic element of any expert system. In general, the inference engine compares the facts in the base of facts with the knowledge, usually represented by production rules. Production rules are an appropriate knowledge representation for this purpose, because on one hand they are easy to be retrieved from human experts and, on the other hand, are suitable for automated processing with expert systems.

Standard production rules (IF – THEN rules) are consisting of two parts: antecedent (evidence, situation, problem) and consequent (hypothesis, action, solution). The production rules formalized by the statement (Mařík et al., 2004)

$E \rightarrow H$,

where E is an evidence and

H is a hypothesis,

are suitable for inferring in both direct and indirect forms. Direct inference is based on the rule of "modus ponens" which is formalized as follows

$$\frac{E, E \rightarrow H}{H},$$

and means that if evidence E and production rule $E \rightarrow H$ are valid, then hypothesis H is also valid. (Gass and Harris, 2001) describe this forward chaining as an approach to reasoning in which an inference engine determines the effect of current known variable values on unknown variables by applying all rules whose premises can be established as being true.

On the other hand, indirect inference can also be used. Indirect inference is expressed by the rule of "modus tollens" and can formally be written as

$$\frac{\neg H, E \rightarrow H}{\neg E}$$

which means that if the hypothesis H is NOT valid and the rule $E \rightarrow H$ is valid, then the evidence E is also NOT valid. Backward chaining refers to an approach to reasoning in which an inference engine endeavours to find a value for an overall goal by recursively finding values for subgoals. Examining rule conclusions to identify rules that could possibly establish a value for the goal is important for the effort of finding a value for the immediate goal (Gass and Harris, 2001).

2.2 Knowledge Unit

For the purposes of this work we understand the knowledge unit as an enhanced production rule containing the compound statement in the antecedent and simple statement in the consequent part. As we showed in our previous work (Dömeová et al., 2008), this kind of the production rules meet better the requirements of a problem-oriented representation of an explicit knowledge.

Formally, we suggested to record knowledge unit as (Dömeováet al., 2008):

$KU = \{X, Y, Z, Q\}$,

where X stands for a problem situation,

Y stands for the problem being solved in the X problem situation,

Z stands for the objective of solving the elementary problem,

Q stands for a successful solution of the elementary problem (result).

Even though there is no unique way to create sentences based on the production rules (Kendal and Creen, 2007), we may always express the knowledge unit in the following textual form (Dömeová et al., 2008):

"*If we want to solve an elementary problem Y in the problem situation X to reach the objective Z, then we should apply the solution Q.*"

Another advantage of the knowledge unit concept is that it allows to define unary and operations with knowledge units such as drill down or roll up (unary operations) or merging or decomposition (binary operations, see Dömeová et al., 2008) with no influence to the complexity of the expressions of the knowledge units in natural languages (Rauchova et al., 2014). These operations could serve as the start point for thinking about the modifications of the inference mechanisms.

3 Results and Discussion

3.1 Inference Chain with Knowledge Unit

To be operable, any knowledge unit could be formally expressed as a production rule in the following form:

IF (X and Y and Z) THEN Q.

This allows to define the relationships among elements of the knowledge units within the basic inference method. The basic form of elementary inference is defined as follows:

Let's have two knowledge units

$KU_1 = \{X_1,Y_1,Z_1,Q_1\}$

and

$KU_2= \{X_2,Y_2,Z_2,Q_2\}$.

Further suppose that the statement IF (X and Y and Z) THEN Q is valid. Then the inference with knowledge units can proceed from element $Q_1$to Y_2, thus

$KU^{INF} = \{Q_1\} ==> \{Y_2\}$

In the language form the operation could be written as follows:

"IFweare not able to apply the solution Q_1, it becomes the elementary problem Y_2solved within the problem situation X_2 to reach the objective Z_2; THENthe solution Q_2 should be applied."

The inference mechanism with knowledge unit is described in Fig. 1.

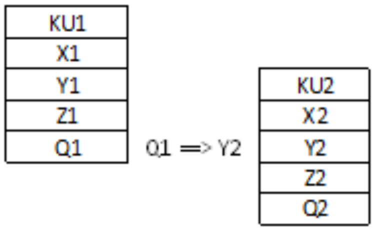

Fig. 1. Inference with knowledge unit

3.2 Case Study

The illustrative example deals with the problem "Application user error analysis" is occurred within the working with an ERP system. The objective is to find errors in the application or find the application, where the error occurred. The error can be vaguely identified by the expert. In this case knowledge units are optimized to knowledge level of an operator on first line helpdesk support. This level was explicitly defined by the expert. For the case study, it is designed for the following scenario:

An error occurred in the application "SharePoint Nintex Workflow" with elementary describe the error as follows: "One column requires a different type of information". This type of error may have origin in several integrated systems. First step is the diagnosis in application where error occurred. The result of first step is the identification application where the error occurred. The second step is diagnosis, which define the module or object, where is the original error in application. And after finding the error is next step to fix the error. The purpose of inference is to find the error in origin application and make first analysis. Boundary of the knowledge domain is the issue of diagnostic errors in the context of integrated applications. Design of the expert system will be similar like in the case of authors (Moreno and Espejo, 2015) and it is diagnostic. As the starting point for the inference the knowledge units developed according to the procedure defined in Houska and Rauchova (2013) is used.

KU1

- X1 Purchase (under the statutes)
- Y1 Create purchase request
- Z1 Provide the delivery for the project
- Q1 Fill out the request form of purchase request and send

KU2

- X2 Solving an error in purchase request workflow
- Y2 Find application, where is error
- Z2 Start analytical test application
- Q2 Identify an error application

We can also KU2 describe: "If we want to solve the error in the “purchase request workflow” and find which applications it is an error, then we must run the analytical test in application where error occurred and identify an error application."

KU3

- X2 Detection error in application
- Y2 Repair error or data inconsistency
- Z2 Ensure correct operation of the system
- Q2 Make a correction in the object

KU4

- X4 Repair in the object KS systemization
- Y4 Detect true values and process consistency
- Z4 Correct classify employee
- Q4 Repair / Correction

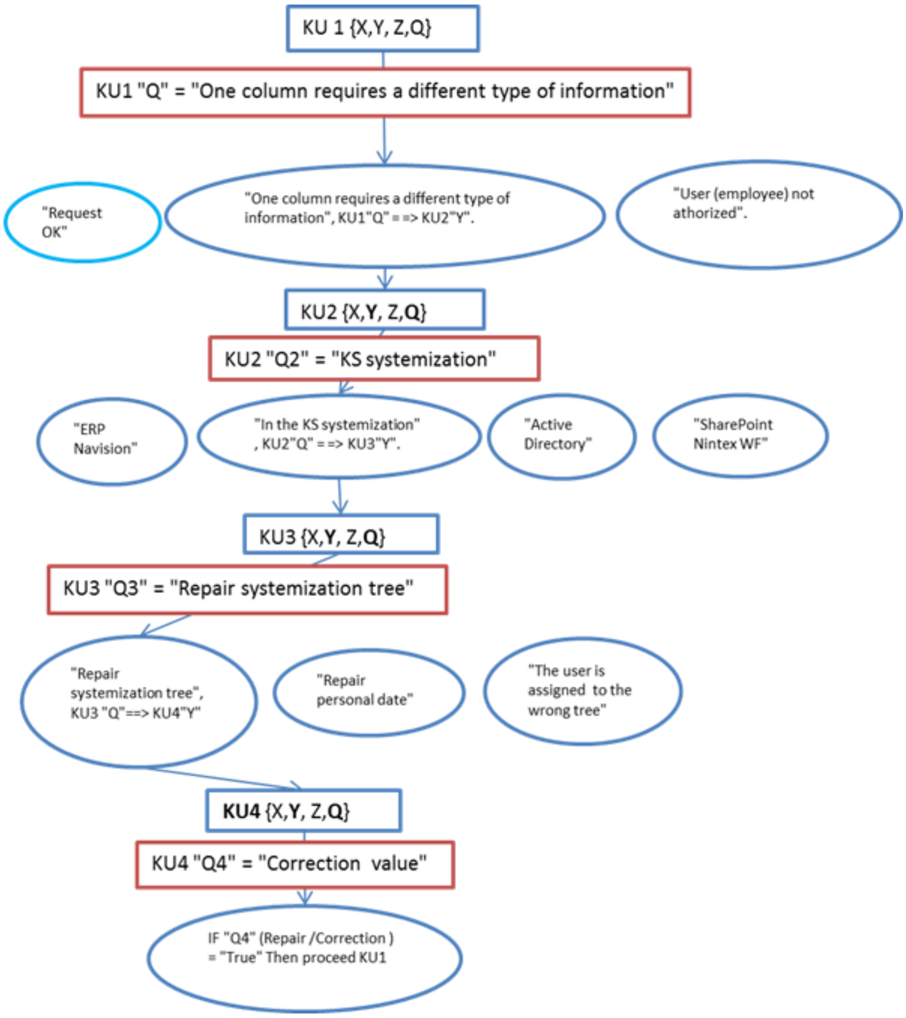

Fig. 2. Inference with knowledge unit in "Application user error analysis".

In this Fig. 2. are knowledge units fill with concrete values during the inference. This case study, demonstrate the inference chaining used for finding errors in application Workflow purchase. The error message stands at the top of the figure. The inference with knowledge units defined location of the error and make result for correction. After correction, continues the process with re-initiating the purchase request. In the KU1, 2, 3 and 4 are concrete knowledge unit for this case:

KU1

- X1 Purchase (under the statutes)
- Y1 Create purchase request
- Z1 Provide the delivery for the project

- Q1 Fill out the request form of purchase request and send = Error state: „One column requires a different type of information”

KU2
- X2 Solving an error in purchase request workflow
- Y2 Find application, where is error ("Q1" = "Y2" ==> "One column requires a different type of information")
- Z2 Start analytical test application
- Q2 Identify an error application = "KS systemization"

KU3
- X2 Detection error in application
- Y2 Repair error or data inconsistency ("Q2" = "Y3" ==> "In the KS systemization)"
- Z2 Ensure correct operation of the system
- Q2 Make a correction in the object = “Repair systemization tree”

KU4
- X4 Repair in the object KS systemization
- Y4 Detect true values and process consistency ("Q3" = "Y4" ==> "Repair systemization tree)
- Z4 Correct classify employee
- Q4 Repair / Correction = "Correction value"

The inference with knowledge unit is defined like special operation between more than one knowledge units. In the real case study is illustrated the inference mechanism with crisp knowledge unit. The case study has a real ground, of the simple problem and it solve problem in few integrated software application in corporation. This case study helps in diagnosis applications problems-errors and is designed for first level of support helpdesk operators.

3.3 Comparison with a Standard Inference Procedure

Base of facts:
- User (employee) not valid, "NAV"
- Systemization number not valid, "KS"
- Wrong data, "ActiveDirectory = AD"
- One column requires a different type of information, "Error"
- Purchase request, "Error", "NAV"
- Systemization, "KS", Wrong number
- Object employee, "NAV", Wrong department
- Employee <=50%
- More AD ID
- Wrong AD tree
- Purchase request, "Error", "AD"
- Purchase request, "Error", "KS"

Base of rules:

- Rule 1 If "User (employee) not valid, "NAV"" Or "Systemization number not valid, "KS"" Or "Wrong data, "AD"" Then "One column requires a different type of information"
- Rule 2 If "Purchase request, "Error", "KS"" Then "User (employee) not valid, "KS""
- Rule 3 If "Systemization, "KS", Wrong number" Then "Purchase request, "Error", "KS""
- Rule 4 If "Purchase request, "Error", "AD"" Then "Wrong data, "AD""
- Rule 5 If “More AD ID" Or "Wrong AD tree" Then "Purchase request, "Error", "AD""
- Rule 6 If "Purchase request, "Error", "NAV"" Then "Systemization, "NAV", Wrong number"
- Rule 7 If "Object employee, "NAV", Wrong department" Then "Purchase request, "Error", "NAV""

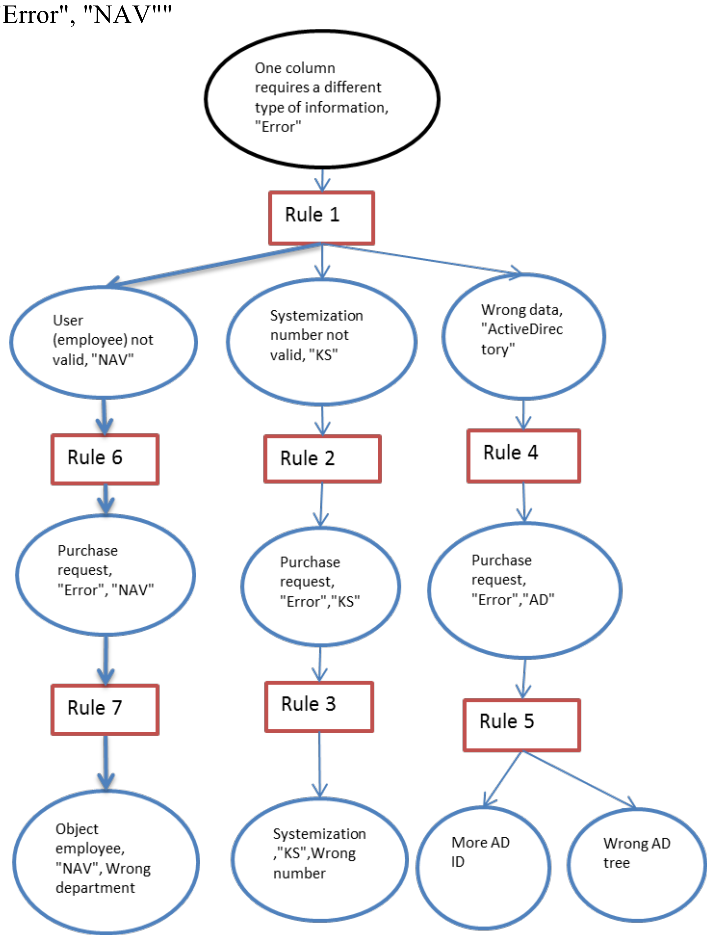

Fig. 3. Standard inference procedure “Backward chaining”

Within the standard inference, the consequent is derived with a sequential chaining of the production rules using the basis of the facts. When inferred knowledge units, individual knowledge units (or their specific parts, respectively) are merged for deriving the consequent, or even another knowledge unit. The main difference between the standard approach and our new one approach lays on the definition of the basis of the facts. In the standard approach, the set of the rules and basis of the facts are strictly separated. On the contrary, knowledge units cover both the production rule and additional information (the problem situation and the solution of the elementary problem (antecedent in a standard production rule)); thus some facts are already integrated directly into the knowledge unit.
Moreover, as the structure of the knowledge units is standardized, it predetermines the order of the knowledge unit by production rules. Just this way of the order allows to base the inference on the facts integrated directly in the units. The inference among two or more knowledge units stems from this principle - the fixed interrelation between the elements of the knowledge units "Q1" ➔ "Y2"

4 Conclusion

In this paper we proposed a way to process a knowledge unit as an enhanced production rule with an inference engine in an expert system. In particular, backward chaining inference algorithm was used. Our approach contributes to decreasing the entropy in the chaining when the production rule consists of compound statements in antecedent or consequent part of the production rule. Of course, the next necessary step is to show the application of our algorithm on more particular examples and compare the success rates of a standard inference algorithm and this new one. The user-oriented point of view could be used for this purpose (Berankova et al., 2008). In case of success new opportunities for future research are opened. First, other inference strategies (e.g. forward chaining) could be modified too to be suitable for inferring the knowledge units. Furthermore, under uncertainty, fuzzy expert systems based on fuzzy knowledge units need the inference strategy for fuzzy knowledge units too.

Acknowledgements. The research is supported by the grant project of the Internal Grant Agency of the FEM CULS Prague “Data, information and knowledge in expert systems”, No. 20171026.

References

1. Aguilera P.A., Fernández A., Fernández R., Rumí R. and Salmerón A. (2011) Bayesian networks in environmental modelling, Environmental Modelling and Software, 26, 12, p.1376-88.

2. Berankova, M., Domeova, L. and Houska, M. (2008) User-oriented Methodology of Communication with Expert Systems. Agricultural Economics-Zemedelska Ekonomika, 54, p.193-201.
3. Chen, S.H. and Pollino, C.A. (2012) Good practice in Bayesian network modelling, Environmental Modelling and Software, 37, p. 134-45.
4. Dömeová, L., Houška, M. and Houšková Beránková, M. (2008) Systems Approach to Knowledge Modelling. Hradec Králové: GSOC.
5. Fakhrahmad, S.M., Sadreddini, M.H. and Zolghadri Jahromi, M. (2015) A proposed expert system for word sense disambiguation: Deductive ambiguity resolution based on data mining and forward chaining. Expert Systems, 32, 2, p.178-91.
6. Gass, S.I. and Harris, C.M. (2001) Encyclopedia of Operations Research and Management Science. US: Springer.
7. Ghanei, S., Vafaeenezhad H., Kashefia M., Eivani A.R. and M. Mazinani M. (2015) Design of an expert system based on neuro-fuzzy inference analyzer for on-line microstructural characterization using magnetic NDT method. Journal of Magnetism and Magnetic Materials, 379, p.131-36.
8. Houska, M. and Rauchova, T. (2013) Methodology of Creating the Knowledge Text. 10th International Conference on Efficiency and Responsibility in Education 2013, p. 197-203.
9. Kendal, S.L. and Creen, M. (2007) An Introduction to Knowledge Engineering. London: Springer.
10. Kim, K.Y. and Kim, Y.S. (2011) Causal design knowledge: Alternative representation method for product development knowledge management. Computer-Aided Design, 43, p.1137-53.
11. Krejci, I. (2013) Age of Machinery and Equipment in Education as Aspect of Modernity. 10th International Conference on Efficiency and Responsibility in Education 2013, pp. 317-23.
12. Mařík, V., Štěpánková, O., Lažanský, J. et al. (2004) Artificial Intelligence I – IV. Prague: Academia Praha.
13. Moreno, C.J. and Espejo, E. (2015) A performance evaluation of three inference engines as expert systems for failure mode identification in shafts. Engineering failure analysis, 53, p. 24-35.
14. Rauchova, T., Houska, M., Luhanova, K., and Cernikova, K. (2014) Comparative Analysis of Quantitative Indicators of Normal and Knowledge Texts. 10th International Scientific Conference on Distance Learning in Applied Informatics (DIVAI 2014), p. 621-631.
15. Venturelli, A., Caputo, F., Leopizzi, R., Mastroleo, G. and Mio, C. (2017) How can CSR identity be evaluated? A pilot study using a Fuzzy Expert System. Journal of Cleaner Production, 141, p.1000-10.