Argument Mining on Italian News Blogs

Pierpaolo Basile
University of Bari
pierpaolo.basile@uniba.it

Valerio Basile
Université Côte d'Azur, Inria,
CNRS, I3S, France
valerio.basile@inria.fr

Elena Cabrio, Serena Villata
Université Côte d'Azur, CNRS,
Inria, I3S, France
firstname.lastname@unice.fr

Abstract

English. The goal of *argument mining* is to extract structured information, namely the *arguments* and their *relations*, from unstructured text. In this paper, we propose an approach to argument relation prediction based on supervised learning of linguistic and semantic features of the text. We test our method on the CorEA corpus of user comments to online newspaper articles, evaluating our system's performances in assigning the correct relation, i.e., *support* or *attack*, to pairs of arguments. We obtain results consistently better than a sentiment analysis-based baseline (over two out three correctly classified pairs), and we observe that sentiment and lexical semantics are the most informative features with respect to the relation prediction task.

Italiano. *L'estrazione automatica di argomenti ha come scopo recuperare informazione strutturata, in particolare gli argomenti e le loro relazioni, a partire da testo semplice. In questo contributo proponiamo un metodo di predizione delle relazioni tra argomenti basato sull'apprendimento supervisionato di feature linguistiche e semantiche del testo. Il metodo è testato sul corpus di commenti di news CorEA, ed è valutata la capacità del sistema di classificare le relazioni di supporto ed attacco tra coppie di argomenti. I risultati ottenuti sono superiori ad una baseline basata sulla sola analisi del sentimento (oltre due coppie di argomenti su tre è classificata correttamente) ed osserviamo che il sentimento e la semantica lessicale sono gli indicatori più informativi per la predizione delle relazioni tra argomenti.*

1 Introduction

The argument mining (Peldszus and Stede, 2013; Lippi and Torroni, 2016) research area has recently become very relevant in computational linguistics. Its main goal is the automated extraction of natural language arguments and their relations from generic textual corpora, with the final goal of providing machine-readable structured data for computational models of argument and reasoning engines. Two main stages have to be considered in the typical argument mining pipeline, from the unstructured natural language documents towards structured (possibly machine-readable) data: *(i)* argument extraction, i.e., to detect arguments within the input natural language texts, and *(ii)* relation extraction, i.e., to predict what are the relations holding between the arguments identified in the first stage. The relation prediction task is extremely complex, as it involves high-level knowledge representation and reasoning issues. The relations between the arguments may be of heterogeneous nature, like attack, support or entailment (Cabrio and Villata, 2013).

The increasing amount of data available on the Web from heterogeneous sources, e.g., social network posts, forums, news blogs, and the specific form of language adopted there challenge argument mining methods, with the aim to support users to understand and interact with such a huge amount of information.

In this paper, we address this issue by presenting an argument relation prediction approach for Italian. We test the method on the CorEA corpus (Celli et al., 2014) of user comments to the news articles of an Italian newspaper, annotated with *agreement* (i.e., support) and *disagreement* (i.e., attack) relations. We extract argument-level features from the CorEA comment (i.e., argument)

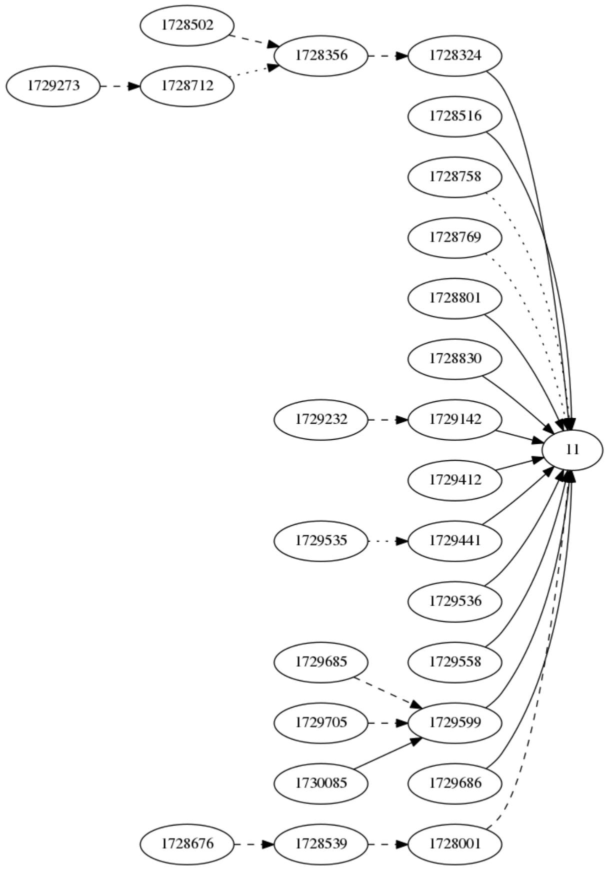

Figure 1: Example of debate structure.

pairs, and we train our system to predict the support and attack relations.

2 Mining Arguments

A debate, whether it happens online or in person, can be modeled as a set of *arguments* proposed by the *participants*. Arguments can be independent, for instance expressing the participant's stance on a particular topic, but often they are replies to previous arguments put forward in the debate. This results in a network structure of the debate, that is, a (possibly disconnected) directed graph where nodes are arguments, and the two kinds of edges are the support and attack relations between them. In Figure 1, each node represents an argument with a numeric identifier, filled and dashed edges represent respectively support and attack relations, and dotted edges are neutral relations. The hub-like node labeled 11 is a news article, thus attracting many first-level comments.

The goal of our work is to be able to predict the relations between the arguments in a given debate, thus reconstructing the relation graph. We therefore cast the problem as a classification task: given two arguments from a debate, we aim to predict whether one argument attacks the other, supports it, or there is no relation between the two arguments. The construction of the graph structure is then straightforward, resulting from the combination of all the argument pairs we considered.

2.1 Features

We extract argument-level features from the CorEA comment pairs, that we group into the following categories:

Lexical We take into account several lexical features: tokens, bi-grams, and the first bi-gram and tri-gram of each argument.

Syntactic We exploit the output of a dependency parser. We consider two kinds of dependency features: the former is the original output, the latter generalizes a word to its POS tag. For instance, "amod(denaro, pubblico)" is generalized as the "amod(NN, pubblico)" and "amod(denaro, ADJ)". We adopt the Malt parser (Nivre, 2003) trained on the Universal Dependency Treebank[1].

Message info We extract the argument size, the number of uppercase words, the number of negations[2], the number of sequences of two or more punctuation characters, the number of citations. A citation is a quoted sequence of words in the second argument that occurs in the first argument.

Message overlap Cosine similarity between two arguments is computed exploiting TF/IDF.

Word-embedding We build word-embeddings relying on the Paisà corpus through the word2vec (Mikolov et al., 2013) tool. We use a vector dimension equal to 50, and we consider only words that occur at least 20 times. For each argument, we use the vector components as features directly.

Sentiment We extract the sentiment from the arguments with two separate tools. Alchemy API[3], the sentiment analysis feature of IBM's Semantic Text Analysis API, returns a polarity label (positive, negative or neutral) and a

[1] `http://universaldependencies.org/it/overview/introduction.html`
[2] The occurrences of the word "non"
[3] `http://www.alchemyapi.com/`

polarity score between -1 (totally negative) and 1 (totally positive). The UNIBA system (Basile and Novielli, 2014), one of the most successful participants in the Sentipolc task at Evalita 2014 (Basile et al., 2014), returns a subjectivity label (subjective or objective) and a polarity label (positive, negative, neutral or mixed).

Topic model We train a domain-independent topic model for Italian and compute, for each argument, its representing vector in the topic space. The 300-dimensional topic model is created with Gensim[4] using the ItWaC corpus (Baroni et al., 2009). We use the vector components as features directly, i.e., each comment has 300 topic-based features.

3 Evaluation

The goal of the evaluation is twofold: *i)* to compute the performance of several machine learning methods and compare them with respect to some baselines, and *ii)* to investigate the importance of each group of features through an ablation test.

3.1 Data

We test our approach on the CorEA corpus (Celli et al., 2014), a collection of text from Italian news blogs. It contains 27 news articles, about 1,660 unique authors and more than 2,900 comments. The corpus is annotated with emotions and, most interestingly for our work, the comments are annotated pair-wise with agreement information (Celli et al., 2016). We extracted such comment pairs for a total of 1,275 pairs: 682 disagreement, 106 neutral, 180 agreement (307 pairs are not classified, examples in Figure 2).

The CorEA dataset provides several information about each message. Beside the features described in Section 2.1, we also extract the following dataset-dependent features: the set of manually annotated topics, the news category of the article, the count of replies to the message, the count of message likes, the participant's activity score, the participant's interests, the participant's page views, the participant's total comments, the participant's total shares, the participant's likes received, and the overall emotion declared by the participant after reading the articles.

[4]https://radimrehurek.com/gensim/

3.2 System setup

We exploit two kinds of learning algorithms: 1) different configurations of SVM based on linear kernel (SVM_{lin}), degree-2 polynomial kernel (SVM_{poly}), and RBF kernel (SVM_{rbf}); 2) Random Forest (RF).

The baseline method always predicts the most frequent class, in this case "attack". Moreover, we test the two simple sentiment analysis systems already described in 2.1, $SA_{alchemy}$ and SA_{uniba}. In particular, these systems exploit the result of the sentiment analysis in terms of polarity (positive, negative, or neutral) for predicting the relation between two arguments: if two arguments have the neutral polarity, they are tagged as neutral, while they are tagged as "support" in case they have the same polarity, otherwise the "attack" class is predicted. The system is implemented in JAVA relying on the Weka tool (Hall et al., 2009). All the experiments are performed by adopting the 10-folds cross-validation. For all the learning methods, we adopt the default Weka parameters since the goal of our work is not to optimize the classification performance but to provide a features study.

3.3 Results

Table 1 reports on the best results obtained by each method. Regarding RF the best result is obtained using 10 trees, while for SVM we optimize only the C parameter using default values for the other ones. The best C value for SVM_{lin} is 1, 2 in all the other settings.

Each one of the supervised systems performs better than the baseline. The good performance of the linear kernel classifier is likely to be ascribed to the high number of features. The performance of Random Forest is also quite good, considering that only ten trees are employed.

Table 1: Results

System	**P**	**R**	**F**
baseline	0.4964	0.7045	0.5824
$SA_{alchemy}$	0.3553	0.3616	0.3584
SA_{uniba}	0.2942	0.3286	0.3105
SVM_{lin}	0.6789	0.7169	**0.6719**
RF	0.6607	0.7180	0.6491
SVM_{poly}	0.6609	0.7097	0.6486
SVM_{rbf}	0.6414	0.7076	0.6120

As can be seen from the results of ablation tests (see Table 2), the features that contribute the most

Relation	Example
Attack	"in certi paesi 100 sterline a settimana permettono di vivere come un pascià" "si ma in certi altri no..;-) la cifra mi sembra davvero esigua.."
Support	"Caro Renzi , hai visto com'è semplice restituire i soldi? Basta una firmetta... perchè non lo fai anche tu invece di promettere e promettere e promettere?" "Bisogna prendere atto che il movimento 5 stelle sta davvero restituendo i soldi agli Italiani. Questo è un fatto, tutto il resto sono chiacchere."
Neutral	"E le riforme?" "le riforme cominciano dl'atteggiamento dei parlamentari. con il cambiamento del mind-set . il punto di partenza."

Figure 2: Examples of relations between pairs of comments in CorEA.

to the argument classification task are the semantic features (i.e., embeddings) and the sentiment features. This confirms our hypothesis that sentiment is a key information for argument mining, and more specifically for the relation prediction task. The results also confirm that lexical and semantic features are useful for the task, as expected. Table 2 reports also the number of features (Feat.Size) and the F1 (F1-f) achieved by exploiting the respective feature in isolation. It is important to note that, despite the bad performance obtained by both embedding and sentiment features, their contribution in the overall performance is relevant.

Table 2: Ablation test

Features	F1	Δ%	Feat.Size	F1-f
all	0.6719	-	220,499	-
-lexical	0.6624	-1.42	140,443	0.66
-syntactic	0.6702	-0.26	80,909	0.65
-info	0.6691	-0.42	220,490	0.58
-CorEA	0.6674	-0.68	220,218	0.64
-embedding	0.6525	-2.89	220,399	0.59
-overlap	0.6724	0.07	220,498	0.58
-sentiment	0.6622	-1.45	220,491	0.58
-topic	0.6673	-0.69	220,045	0.59

4 Related Work

(Lippi and Torroni, 2016) and (Peldszus and Stede, 2013) provide an overview about the argument mining research area. In particular, some approaches have been recently proposed to address the same task addressed in this paper, i.e. predicting relations between arguments, even if ours is the first effort for the Italian language. (Aharoni et al., 2014) assume that evidence is always associated with a claim, enabling the use of information about the claim to predict the evidence. The support relations are thus obtained by definition when predicting the evidence. (Mochales and Moens, 2011) have addressed the problem by parsing with a manually-built context-free grammar to predict relations between argument components. The grammar rules follow the typical rhetorical and structural patterns of sentences in juridical texts. This is a highly genre-specific approach, and its direct use in other genres would be unlikely to yield accurate results. (Stab and Gurevych, 2014) instead employ a binary SVM classifier to predict relations in a claim/premise model. (Biran and Rambow, 2011) apply the same method adopted for the detection of premises also for the prediction of relations between premises and claims. (Wang and Cardie, 2014) apply an isotonic Conditional Random Fields based sequential model to make predictions on sentence- or segment-level on discussions on Wikipedia Talk pages. Finally, (Cabrio and Villata, 2013) adopt Textual Entailment to infer whether a support or attack relation between two given arguments holds.

5 Conclusions

In this paper, we have presented a supervised approach for argument relation prediction for Italian, mainly relying on features including semantics and sentiment. We tested such approach on the CorEA corpus, extracted from user comments to online news. Our experimental results are good, and foster future research in the direction of including semantics as well as sentiment analysis in the argument mining pipeline. It will be also interesting, as future work, to refine the model in order to consider the full sequence of interactions between arguments.

References

Ehud Aharoni, Anatoly Polnarov, Tamar Lavee, Daniel Hershcovich, Ran Levy, Ruty Rinott, Dan Gutfreund, and Noam Slonim. 2014. A benchmark dataset for automatic detection of claims and evidence in the context of controversial topics. In *Proceedings of the First Workshop on Argumentation Mining*, pages 29–38, Baltimore, Maryland, June. Association for Computational Linguistics.

Marco Baroni, Silvia Bernardini, Adriano Ferraresi, Eros Zanchetta, Springer, and Science+business Media B. V. 2009. The wacky wide web: A collection of very large linguistically processed webcrawled corpora. language resources and evaluation.

Pierpaolo Basile and Nicole Novielli. 2014. Uniba at evalita 2014-sentipolc task: Predicting tweet sentiment polarity combining micro-blogging, lexicon and semantic features. *Proceedings of EVALITA*, pages 58–63.

Valerio Basile, Andrea Bolioli, Malvina Nissim, Viviana Patti, and Paolo Rosso. 2014. Overview of the Evalita 2014 SENTIment POLarity Classification Task. In *Proceedings of the 4th evaluation campaign of Natural Language Processing and Speech tools for Italian (EVALITA'14)*, Pisa, Italy.

Or Biran and Owen Rambow. 2011. Identifying justifications in written dialogs by classifying text as argumentative. *Int. J. Semantic Computing*, 5(4):363–381.

Elena Cabrio and Serena Villata. 2013. A natural language bipolar argumentation approach to support users in online debate interactions†. *Argument & Computation*, 4(3):209–230.

Fabio Celli, Giuseppe Riccardi, and Arindam Ghosh. 2014. Corea: Italian news corpus with emotions and agreement. In *CLIC-it 2014*, pages 98–102.

Fabio Celli, Giuseppe Riccardi, and Firoj Alam. 2016. Multilevel annotation of agreement and disagreement in italian news blogs. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may. European Language Resources Association (ELRA).

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The weka data mining software: An update. *SIGKDD Explor. Newsl.*, 11(1):10–18, November.

Marco Lippi and Paolo Torroni. 2016. Argumentation mining: State of the art and emerging trends. *ACM Trans. Internet Techn.*, 16(2):10.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *Workshop at ICLR, 2013*.

Raquel Mochales and Marie-Francine Moens. 2011. Argumentation mining. *Artificial Intelligence and Law*, 19(1):1–22.

Joakim Nivre. 2003. An efficient algorithm for projective dependency parsing. In *Proceedings of the 8th International Workshop on Parsing Technologies (IWPT)*.

Andreas Peldszus and Manfred Stede. 2013. From argument diagrams to argumentation mining in texts: A survey. *IJCINI*, 7(1):1–31.

Christian Stab and Iryna Gurevych. 2014. Identifying argumentative discourse structures in persuasive essays. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 46–56.

Lu Wang and Claire Cardie. 2014. Improving agreement and disagreement identification in online discussions with a socially-tuned sentiment lexicon. In *Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 97–106, Baltimore, Maryland, June. Association for Computational Linguistics.