A Machine Learning Approach to Argument Mining in Legal Documents

Prakash Poudyal(✉)

Department of Informatics, University of Évora, Évora, Portugal
prakashpoudyal@gmail.com

Abstract. This study aims to analyze and evaluate the natural language arguments present in legal documents. The research is divided into three modules or stages: an Argument Element Identifier Module identifying argumentative and non-argumentative sentences in legal texts; an Argument Builder Module handling clustering of argument's components; and an Argument Structurer Module distinguishing argument's components (premises and conclusion). The corpus selected for this research was the set of Case-Laws issued by the European Court of Human Rights (ECHR) annotated by Mochales-Palau and Moens [8]. The preliminary results of the Argument Element Identifier Module are presented, including its main features. The performance of two machine learning algorithms (Support Vector Machine Algorithm and Random Forest Algorithm) is also measured.

Keywords: Legal argument · Natural language analysis
Machine learning

1 Introduction

An argument combines a premise or a set of premises and a conclusion. Historically, Dialectics and Philosophy are the ancient roots of the discipline of argumentation. Arguments have always been considered an important branch of Philosophy and, with the passage of time and advancement in technology, its relevance has grown exponentially in other fields such as Literature, Logic, Law, and also in Mass Communication and Artificial Intelligence. Arguments are the fundamental tools for human beings to argue and reach their objectives. During debates, the conclusion of an argument is the focal point of the discussion. Premises are the vehicle that supports the conclusion's reasoning and approval. There are premises that reinforce other premises and as such add strength to the conclusion. During a discussion, facts, figures and further evidence as well as logic are provided to support, attack and/or refute the opponent's arguments. At a time when social media is one of the most important discussion platforms available, the number of users expressing their opinion has grown exponentially. Usually, such opinions are expressed through an array of premises that generate ideas and claims. Considering the relevance of argumentation in everyday life and its ubiquity in the

U. Pagallo et al. (Eds.): AICOL VI-X 2015–2017, LNAI 10791, pp. 443–450, 2018.
https://doi.org/10.1007/978-3-030-00178-0_30

judiciary, this study was made to analyse and evaluate the natural language used in argumentative legal documents. To automatically identify the argument in an unstructured text, a system was developed in three stages or modules. The first stage or module is the Argument Element Identifier, henceforth referred to by its acronym AEI. In this module, the main aim was to identify the argumentative and non-argumentative sentences in a corpus of legal documents. The structuring of arguments is addressed in the second stage or the Argument Builder Module, henceforth referred to as AB. In the third stage, the Argument Structurer Module (henceforth referred to as AS), the system will distinguish the arguments' components (premise and conclusion). The corpus selected for this study was the Case-Law issued by the European Court of Human Rights (ECHR) annotated by Mochales-Palau and Moens [8]. Details of the corpus are described in [11].

Mochales-Palau and her colleagues [6–10,13] have published several papers identifying and extracting arguments from both the ECHR Corpus and the Araucaria Corpus[1]. Moens *et al.* [9] used features such as n-gram, verb nodes, word couples, and punctuation and their average accuracy results was close to 74% in various types of text but dropped slightly to 68% in the legal corpus. Mochales-Palau and Moens [8] added more features such as modal auxiliary, keywords, negative/positive words, text statistics, punctuation keywords, same words in both the previous as well as the following sentence, and first and last words in the next sentence and reported accuracy results of 90%. Mochales-Palau and Moens [10] also defined the argument boundaries i.e. the beginning as well as the end of an argument. Since components of the argument can be found scattered throughout the text, the authors suggest using semantic distance to solve this issue and argue for the use of context-free grammars (CFG) to detect the argument structure and claim to have reached and accuracy of 60%. The technique presented by these authors is applied only to a very limited number of Case-Laws.

Stab *et al.* [15,16] analysed argumentative writings from a discourse structure perspective. They used structural, lexical, syntactic and contextual features to determine argumentative discourse structures in persuasive essays. Their experiment succeeded in establishing the f-measure for identifying argument components at 0.726. They focused on word indicators and lexical features that highlight an argumentative sentence. Doddington *et al.* [4] described four challenges and identified five types and 24 subtypes of relations. The "Role" type of relation, which refers to the part a person plays in an organization, can be subtyped as Manager, General Staff, Member, Owner, Founder, Client, Affiliate-Partner, Citizen-of or Other. The "Part" type of the relation can be subtyped as Subsidiary, Part-of or Other. The "Near" type identifies relative locations. The "Social" type can be subtyped as Parent, Sibling, Spouse, Grandparent, Other-Relative, Other-Personal, Associate, or Other-Professional.

Bunescu and Mooney [2] presented a novel approach to extract the relation between entities by presenting a new kernel for the relation extraction, based on the shortest path between the two relation entities in a dependency graph. They

[1] http://araucaria.computing.dundee.ac.uk/doku.php.

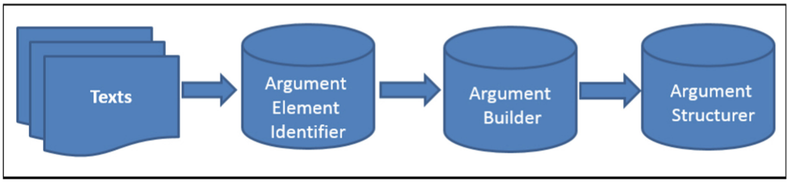

Fig. 1. Proposed Architecture of the System

deployed an “Automatic Content Extraction” on a corpus of newspaper articles and were able to show significant improvements over a recent dependency tree kernel. Biran and Rambow [1] also aimed to identify argumentative relations while Cabrio and Villata *et al.* [3] used a combination of textual entailment framework and bipolar abstract argumentation approach to evaluate argument texts and find the relation between the arguments. Florou *et al.* [5] used a grammatical approach of future and conditional tenses and moods. They highlight the impact of illustration, justification, and rebuttal wording in the argument. Poudyal and Quaresma [12] have found that the Support Vector Machine is the best machine learning algorithm in identifying name entity relation.

2 Proposed Approach

The system we propose consists of three sequential modules or phases as illustrated by Fig. 1.

1. Argument Element Identifier (AEI): identifies argumentative and non - argumentative sentences in legal texts;
2. Argument Builder (AB): handles arguments’ components’ clustering;
3. Argument Structurer (AS): distinguishes arguments components (premise and conclusion).

During the Argument Element Identifier (AEI) phase, our main task was to find an optimal machine learning algorithm with appropriate features to distinguish an argumentative from a non-argumentative sentence in legal documents. We conducted several experiments with various machine learning algorithms and classified them according to the type of features used. Figure 2 presents an overview of the AEI phase. After identifying the argumentative sentences in a legal text, it is necessary to organize these sentences into argumentative clusters composed by a set of argumentative sentences interconnected or related to each other. Detecting the boundaries of an argument is a very challenging task mainly due to the fact that its components (premise and conclusion) may be connected or related to other arguments. To cluster such sentences, we deployed a fuzzy clustering algorithm (FCA) that provides a membership value ranging from 0 to 1 for each sentence cluster. The membership values are the key assets of the FCA, which allows us to associate each sentence to more than one argument cluster. The performance of the algorithm depends on the type of features

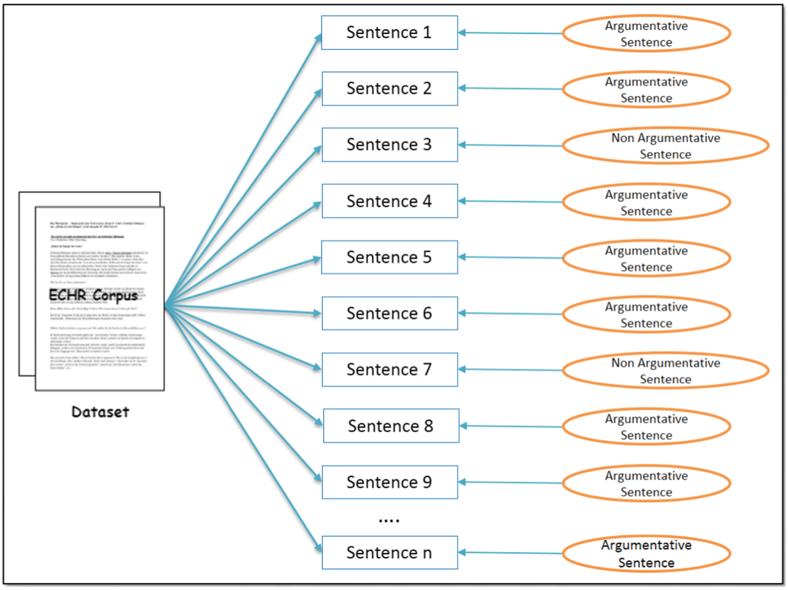

Fig. 2. Overview of the Argument Element Identifier Module (AEI)

selected. In this study, we focus on the following features: 'N-gram', 'Word2vec', and 'Sentence Position'. Figure 3 offers an overview of this phase. On the AS phase, argument components (premise and conclusion) are identified as having a premise or a conclusion basis. The sentences identified as having a premise basis are outright premises or consist of a premise clause. The sentences identified as having a conclusion basis are obvious conclusions or point towards one. Many sentences that have a premise basis and are tagged as such may also include a conclusion clause and the same happens to the sentences labeled as displaying a conclusion basis. To accomplish this task, we deployed indicator features. Indicator features play an important role in identifying argument's premises and conclusions. Words such as "finally," "therefore," "concluding," and "thus" clearly introduce a conclusion and play an important role in the process of identifying argument's conclusions. It is also highly probable that sentences containing words like "should," "could," "almost," "must be," "because," "seems," and "would like," are premises. A major limitation in the AS phase is that each sentence may have one or several premises but only one conclusion, and also the system's accuracy rate will diminish whenever the classifier is not able to identify the sentence's conclusion, or identifies more than one conclusion in a single argument.

3 AEI Preliminary Results

The main goal of the AEI phase was to select the algorithm with the most appropriate parameters. We aimed to develop a system that will automatically

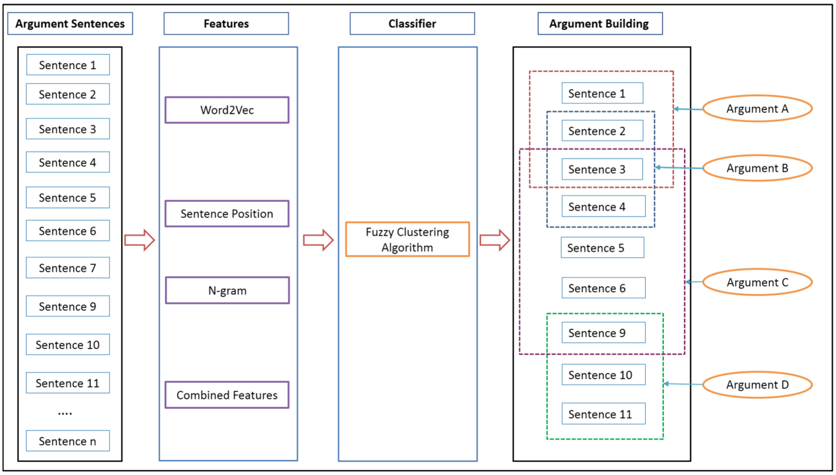

Fig. 3. Overview of the Argument Builder (AB) Module

identify the argumentative sentences on an unstructured textual document. As Fig. 4 illustrates, the AEI system's architecture follows several steps. Initially, the corpus needs to be refined. Once the features are extracted, the classifier can then be built and its performance evaluated.

The words that form a document must be mapped in accordance to a predetermined token and TF-IDF in order to normalise the length of each unit. In our experiment, this procedure created 11374 features. The TF-IDF [11] function was calculated as:

$$tf - idf(w_i, d) = tf(w_i, d)ln\frac{N}{df(w_i)} \tag{1}$$

where $tf(w_i d)$ is the frequency word w_i in document d and $df(w_i)$ is the number of documents where w_i appears and N is the number of documents in the corpus. To measure performance we used precision, recall and f-measure [14] methods. We ran several experiments with the machine learning algorithms Support Vector Machine (SVM) and Random Forest (RF) to determine their performance in identifying argumentative sentences in accordance with the features provided. We selected the top-n informative features (using the gain ratio measures) with $n \in \{100, 200, 500, 1000, 2000, 5000, 11374\}$ and tested the polynomial kernel SVM with various values for the complexity parameter ($C \in \{0.001, 0.01, 0.1, 1, 10,$ and $100\}$). Similar experiments were conducted deploying the Random Forest algorithm using several trees ($nt \in \{7, 11, 17, 50, 100\}$).

Figures 5 and 6 show the graph of f-measure vs. Support Vector Machine (SVM) algorithm and f-measure vs. Random Forest Algorithm respectively. In the SVM chart (Fig. 5), as the number of features increases, the performance of f-measure increases, up to 2000 features. The highest f-measure value of 0.595 was achieved with c = 0.1 and 2000 features in the SVM algorithm experiment.

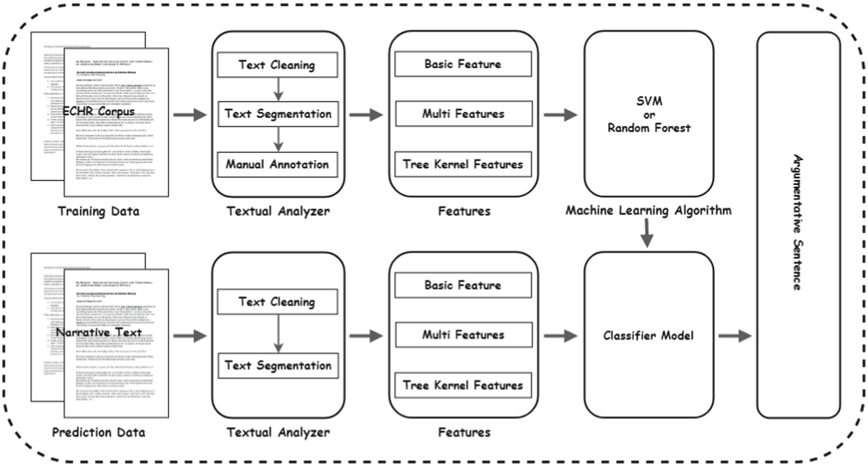

Fig. 4. Architecture of argument element identifier (AEI) Module

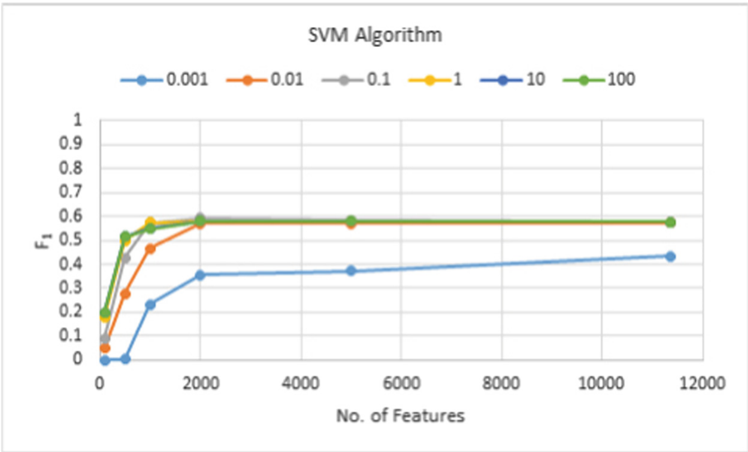

Fig. 5. F-Measure of SVM algorithm

In case of the graph of f-measure obtained from the Random Forest Algorithm chart, (Fig. 6) as the number of features increases, a peak f-measure of 0.52 was reached with 1000 features and 100 trees. Then, the f-measure value decreases up to 2000 and remains constant till 11681 features. We can therefore conclude that the SVM algorithm produced better results than the RF algorithm. Overall, the results achieved are quite promising and support our proposal for the creation of a new argument mining framework.

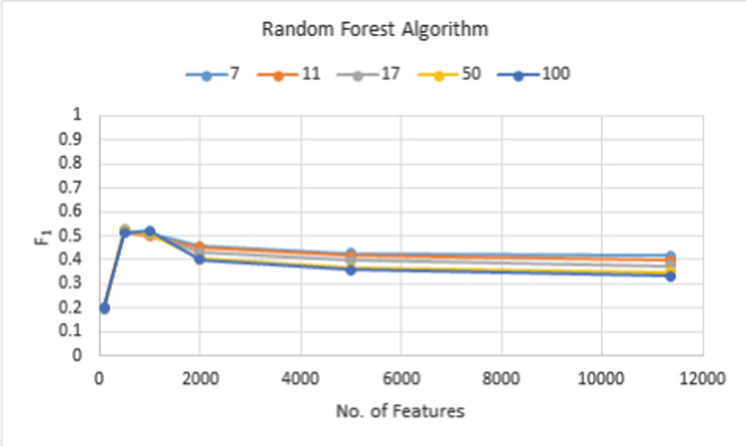

Fig. 6. F-Measure of RF algorithm

4 Conclusion and Future Works

We are proposing a new approach to automatically identify arguments in legal documents which is phased in three modules: Argument Element Identifier (AEI), Argument Builder (AB) and Argument Structurer (AS). The preliminary results of the AEI are extremely promising and support to the development of a new argument mining framework. Further research must be done on the use of string kernel as well as other alternative representation models, including linguistic features such as POS tags, Parse trees and Tree Kernel.

Acknowledgment. The current work is funded by EMMA-WEST in the framework of the EU Erasmus Mundus Action 2.

References

1. Biran, O., Rambow, O.: Identifying justifications in written dialogs by classifying text as argumentative. Int. J. Semant. Comput. **5**(04), 363–381 (2011). https://doi.org/10.1142/S1793351X11001328
2. Bunescu, R.C., Mooney, R.J.: A shortest path dependency kernel for relation extraction. In: Proceedings of the Human Language Technology Conference and Conference Empirical methods in Natural Language Processing (HLT/EMNLP-05), pp. 724–731. Association for Computational Linguistics, Stroudsburg (2005). https://doi.org/10.3115/1220575.1220666
3. Cabrio, E., Villata, S.: Towards a benchmark of natural language arguments. In: Proceedings of the 15th International Workshop on Non-Monotonic Reasoning (NMR 2014), Vienna (2014)
4. Doddington, G., Mitchell, A., Przybocki, M., Ramshaw, L., Strassel, S., Weischedel, R.: The automatic content extraction (ace) program-tasks, data, and

evaluation. In: Proceedings of the Fourth International Conference on Language Resources and Evaluation, vol. 2, pp. 837–840 (2004)
5. Florou, E., Konstantopoulos, S., Koukourikos, A., Karampiperis, P.: Argument extraction for supporting public policy formulation. In: Proceedings of the 7th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities, pp. 49–54 (2013)
6. Mochales, R., Ieven, A.: Creating an argumentation corpus: do theories apply to real arguments?: a case study on the legal argumentation of the ECHR. In: Proceedings of the 12th International Conference on Artificial Intelligence and Law, pp. 21–30. ACM, New York (2009). https://doi.org/10.1145/1568234.1568238
7. Mochales, R., Moens, M.F.: Study on the structure of argumentation in case law. In: Proceedings of the 2008 Conference on Legal Knowledge and Information Systems, pp. 11–20. IOS Press, Amsterdam (2008)
8. Mochales-Palau, R., Moens, M.F.: Study on sentence relations in the automatic detection of argumentation in legal cases. Front. Artif. Intell. Appl. **165**, 89–98 (2007)
9. Moens, M.F., Boiy, E., Palau, R.M., Reed, C.: Automatic detection of arguments in legal texts. In: Proceedings of the 11th International Conference on Artificial Intelligence and Law, pp. 225–230. ACM (2007)
10. Palau, R.M., Moens, M.F.: Argumentation mining: the detection, classification and structure of arguments in text. In: Proceedings of the 12th International Conference on Artificial Intelligence and Law, pp. 98–107. ACM (2009). https://doi.org/10.1145/1568234.1568246
11. Poudyal, P., Goncalves, T., Quaresma, P.: Experiments on identification of argumentative sentences. In: Proceeding of 10th International Conference on Software, Knowledge, Information Management & Applications (SKIMA), pp. 398–403. IEEE (2016). https://doi.org/10.1109/SKIMA.2016.7916254
12. Poudyal, P., Quaresma, P.: An hybrid approach for legal information extraction. Front. Artif. Intell. Appl. (JURIX) **250**, 115–118 (2012). https://doi.org/10.3233/978-1-61499-167-0-115
13. Reed, C., Palau, R.M., Rowe, G., Moens, M.F.: Language resources for studying argument. In: Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC 2008), Marrakech, Morocco, pp. 91–100 (2008)
14. Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic indexing. Commun. ACM **18**(11), 613–620 (1975). https://doi.org/10.1145/361219.361220
15. Stab, C., Gurevych, I.: Identifying argumentative discourse structures in persuasive essays. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 46–56 (2014). https://doi.org/10.3115/v1/D14-1006
16. Stab, C., Kirschner, C., Eckle-Kohler, J., Gurevych, I.: Argumentation mining in persuasive essays and scientific articles from the discourse structure perspective. In: Proceedings with the Workshop on Frontiers and Connections between Argumentation Theory and Natural Language Processing, Bertinoro, Italy, pp. 40–49 (2014)