A Stacking Approach for Cross-Domain Argument Identification

Conference Paper · August 2021

DOI: 10.1007/978-3-030-86472-9_33

6 authors, including:

Alaa Alhamzeh
Institut National des Sciences Appliquées de Lyon
9 PUBLICATIONS **33** CITATIONS
SEE PROFILE

Mohamed Bouhaouel
Universität Passau
3 PUBLICATIONS **12** CITATIONS
SEE PROFILE

Jelena Mitrović
Universität Passau
70 PUBLICATIONS **599** CITATIONS
SEE PROFILE

Harald Kosch
Universität Passau
282 PUBLICATIONS **2,013** CITATIONS
SEE PROFILE

A Stacking Approach for Cross-Domain Argument Identification

Alaa Alhamzeh[1,2], Mohamed Bouhaouel[2], Előd Egyed-Zsigmond[1], Jelena Mitrović[2], Lionel Brunie[1], and Harald Kosch[2]

[1] INSA de Lyon, 20 Avenue Albert Einstein, 69100 Villeurbanne, France
[2] Universität Passau, Innstraße 41, 94032 Passau, Germany
{Alaa.Alhamzeh, Elod.Egyed-zsigmond, Lionel.Brunie}@insa-lyon.fr,
{Mohamed.Bouhaouel, Jelena.Mitrovic, Harald.Kosch}@uni-passau.de

Abstract. Argument identification is the cornerstone of a complete argument mining pipeline. Furthermore, it is the essential key for a wide spectrum of applications such as decision making, assisted writing, and legal counselling. Nevertheless, most existing argument mining approaches are limited to a single, specific domain. The problem of building a robust system whose models are able to generalize over heterogeneous datasets remains fairly unexplored. In this paper, we tackle the argument identification task on two different datasets (Student Essays and Web Discourse), following two approaches: a classical machine learning approach and a DistilBert-based approach. Moreover, this paper sheds light on a new direction for researchers in this domain since we validate the principle of ensemble learning. In other words, we show that combining multiple approaches via a well stacked model improves the system performance. The results are very promising with respect to the recent findings in the literature.

Keywords: Argument Mining · Argument Identification · Computational Linguistics · Classical Machine Learning · Transfer Learning · Stacking.

1 Introduction

Argumentation is a fundamental aspect of human communication, thinking, and decision making. It can be defined as the logical reasoning humans use to come to a conclusion or justify their opinions on a specific topic. It was first studied by ancient Greek philosophers in 6th century B.C., and they are known today as the first argumentation theorists. Later on, argumentation gained more attention from different domains like psychology, communication, linguistics and, more recently, computer science, in particular, as a Natural Language Processing (NLP) task.

An argument consists of two elementary components: one or more premises supporting one claim(conclusion) [1]. According to Missimer [2], *"The objective of argumentation is to convince an opponent of a certain* ***claim****. The claim is a*

perspective or belief that is justified through logical reasoning. The reasoning is an ***inference relation*** *drawn from* ***supporting evidence*** *or reasons towards the claim. If the reasoning is valid, then the claim is a legitimate conclusion of the provided reasons. The manifestation of the application of this process is called an* ***argument****."*

Keeping this definition in mind, we can see the need for different argument diagrams and schemes. Each domain expert looks at the argument and the inference structure from a different angle considering the requirements of the task at hand. Thus, he tries to represent the relations between the premises and claims using a relative scheme. Actually, that leads to one of the main challenges in this domain - the problem of different annotation practices of available datasets, which we elaborate on in Section 2. Consequently, most studies have been concentrating only on one individual subtask of the following:

- Argument Identification: classification of text into Argument or non-argument.
- Argument Components classification: detection of premises and claims.
- Argument structure identification: consists of the argument components plus the relations between them [3].

Argument identification is, therefore, the first essential block in an argument mining pipeline. This phase is important because not every sentence in a text is a part of an argumentation process (some narrative parts serve as introduction or summary about the topic and are not relevant for the argument itself). Since a low performance in argument identification would eventually propagate further down to the next tasks, this step is currently the most refined subtask of argument mining and it will be our focus in this paper.

State-of-the-art literature reports mostly classical machine learning models and very few attempts at using deep learning to solve this problem[4–6].

In this paper, we strive to achieve the maximum profit of both approaches by combining a classical Support Vector Machine(SVM) model [7] with a DistilBERT based model [8], using the concept of ensemble learning [9], which to the best of our knowledge has not been used yet in the domain of argument mining. This is a promising approach due to the fact that it gathers the power of both existing techniques and maximizes the accuracy of the final prediction. On the other hand, it makes a union between the high performance of deep learning models and the interpretability of classical machine learning models.

The contributions of this paper are:

(a) Implementation of a new transfer learning model for Argument Identification based on DistilBert [8], a distilled version of BERT[10]: smaller, faster, and lighter with 97% of its language understanding capabilities [8].
(b) Proposing of an ensemble learning model which combines the out of two different approaches and improve the overall system performance.
(c) We test our individual and ensembled models on two different corpora, and achieved a good improvement by the overall model.
(d) Our work is available publicly through the github repository.[3]

[3] Project source code is available at https://github.com/Alaa-Ah/Stacked-Model-for-Argument-Mining

This paper is organized as follows: in Section2, we take a close look at the conceptual background of our work as well as the state-of-the-art studies considering our particular task of argument identification. In Section3, we come to our contribution details. We validate the results in Section 4. Finally, we discuss the overall research questions and future work in Section 5.

2 Related Work

In this section, we first concentrate on the state-of-the-art classical solutions regarding the argument identification task. Later on, we discuss the concept of transfer learning and come through the different argumentation tasks that it has been evaluated on. Finally, we introduce a conceptual background about ensemble learning and have a deeper look at its essence.

2.1 Argument Mining

The research in this domain witnesses a variance between the period before and after contextualized embeddings and transformers. Indeed, transformers have rapidly become the model of choice for NLP problems [11].

Traditional Machine Learning Methods: The problem of argument identification itself is a binary classification task. Many approaches have been conducted in the literature. Moens et al. [12] worked on detecting arguments from legal text. They investigated a set of textual features (word pairs, text statistics, verb features and keywords) on the Araucaria corpus, which contains arguments from various sources, and achieved the best results with a multinomial naïve Bayes classifier. They found that the lack and the ambiguity of the linguistic markers (e.g. should), seem to be a major source of errors. They revisited the topic in [13] and applied the same method on the ECHR (European Court of Human Rights) corpus as well as to the Araucaria corpus. The classifier performed significantly better on the ECHR corpus, which seems plausible considering that language cues in legal argumentation texts are more explicit and more restrictive. However, the AraucariaDB received additional annotations and modifications over the years. As of today, it does not include the original text anymore like it did in its first version. As a result, we could not use it for the argumentative text identification task.

Stab et al. [14] introduced the annotation of argument components in student essays corpus. They also identified the argumentative discourse structure of this corpus in [15, 3]. Habernal et al. [16] worked on annotating and mining arguments from user-generated Web discourse. In our study, we investigate both of the latter corpora, therefore more details on this process will be stated in Section 3.

However, even an efficient domain-specific model was unable to reach a satisfying result in different domain corpora. Cross-domain identification of argument units is still a rarely tackled problem. One recent attempt is by J.Daxenbereger et al. [17] where the authors tried to overcome the problem of domain dependence

only for claim identification. They found that in-domain and cross-domain experiments have few shared properties on the lexical level (like the word "should").

Transfer Learning Methods: In a traditional machine learning model, there is always an assumption that the training and testing data follow the same distribution and serve the same task. On the contrary, transfer learning seeks freedom from those constraints and searches for methods to adapt models trained on a given dataset to classify slightly different data. It goes towards the search of domain, task, and corpus agnostic models [18]. In principle, it aims to apply previous learned knowledge from one source task (or domain) to a different target one, considering that source and target tasks and domains may be the same or may be different but related.

So why is this useful in the argument mining domain? First of all, common knowledge about the language is obviously appreciable. Second, transfer learning can solve or at least help to solve one of the biggest challenges in the argument mining field, the lack of labeled datasets. Third, even available datasets are often of small size and very domain and task dependent. They may follow different annotations, argument schemes, and various feature spaces. This means that in each potential application for argument mining, we need argument experts to label a significant amount of data for the task at hand, which is definitely an expensive work in terms of time and human-effort. Hence, transfer learning will fine-tune a pre-trained knowledge on a big dataset to serve another problem.

Recently, in 2020, only two studies addressing transfer learning models for argumentation tasks were published. The first one is towards discriminating evidence related to Argumentation Schemes [19] where the authors train classifiers on the sentence embeddings extracted from different pre-trained transformers. The second one is by Wambsganss et al. [4] where the authors proposed an approach for argument identification using BERT (Bidirectional Encoder Representations from Transformers) [10]. Our transfer learning model is based on a distilled version of BERT, proposed by [8], which retains 97% of the language understanding capabilities of the base BERT model, with a 40% less in size, and being 60% faster.

2.2 Ensemble Learning

Ensemble learning is a machine learning research area where different models (i.e. learners) are trained to solve the same problem and combined to get better results [9]. The fundamental hypothesis behind it, is that when different models are correctly combined, the ensemble model tends to outperform each of the individual models in terms of accuracy and robustness [20].

This concept of ensemble learning usually comes to the scene with weak learners, so the overall model is highly improved (e.g., [21, 22]). In our particular case, each individual model provides a considerable performance. Our goal, therefore, is to benefit from all the features a classical ML model uses, and the contextualized knowledge a deep learning model reveals, aiming to improve the performance and stability of the model.

To the best of our knowledge, this promising concept has never been used in argumentation tasks. Hence, we expect this paper to highlight its strength and potential.

3 Contribution

In this section, we present the setup of our experiments in addition to the methods that have been adopted to achieve the goal of extracting argumentative clauses from a natural text.

3.1 Problem Statement

The argument mining problem is broad and can be seen as a set of several sub-tasks. In this paper, we consider argument identification as a flat problem. Thus, a text contains only arguments and non-arguments. In our proposed approach, we decide to do **text classification** at **sentence level** in order to maintain the complete meaning of the sentence and ease the identification. For instance, given an argumentative passage, (1) we first apply sentence segmentation (i.e. split the text into sentences) and then we (2) classify each sentence as argument or non-argument by two individual models which we discuss in Sections 3.3 and 3.4. (3) We finally combine their predictions by a stacked method that is capable to improve the performance on the two corpora as described in Section 3.5.

3.2 Corpora Description

In argument mining, finding a suitable annotated dataset for a specific task is very challenging due to, on one hand, the different scheme annotations of the available labeled datasets on the web. On the other hand, the annotation task itself is expensive. Moreover, labeling or annotating a new corpus needs typically domain experts to validate it. In our particular case, and in order to achieve the goal of cross-domain argument identification model, we searched for datasets that contain both argument and non-argument labels. Student Essays [14] and Web discourse [16] are public corpora which serve this purpose well.

The **Student Essays corpus** contains 402 Essays about 8 controversial topics. The annotation covers the following argument components: 'major claim', 'claim' and 'premise'. Moreover, it presents the support/attack relations between them. Thus, it could be used in several argument mining tasks.

The **User-generated Web Discourse corpus** is a smaller dataset that contains 340 documents about 6 controversial topics in education such as homeschooling. The document may refer to an article, blog post, comment, or forum posts. In other words, this is a noisy, unrestricted and less formalized dataset. The annotation has been done by [16] according to Toulmin's model [23]. Thus, it covers the argument components: 'claim', 'premise', 'backing' and 'rebuttal'.

In order to deduct a binary-labeled unified data for both corpora, we label any argument component as an 'argument', and the rest of the text sentences as 'non-argument'.

3.3 Classical Machine Learning Model - SVM

In terms of the first base model, we consider training a classical machine learning model. This model should be able to capture and learn textual features and patterns that identify argumentative sections of text. Inspired by the works of [15, 12], we defined a set of structural, lexical, and syntactic features in addition to discourse markers as shown in Table 1.

The structural features reflect the building of the sentence and its position in the document. For instance, *tokens count* or length of the sentence exploit the fact that premises tend to be longer than other sentences which can therefore contribute to the argument identification process. Likewise, *question mark ending* indicates that a sentence ending with a question mark is more likely to be a claim, and eventually an argument.

Table 1. Textual Features, new added features are marked with '*'

	Features	**Explanation**
Structural features	sentence position [3]	Indicates the index of the sentence in the document.
	tokens count [3, 12]	Indicates the count of tokens (words) in the sentence.
	question mark ending [3]	Boolean feature.
	punctuation marks count [3]	Indicates how many punctuation marks are there in the sentence.
Lexical features	1-3 gram BoW [3, 12]	Unigrams, bigrams and trigram Bag of Words features.
	1-2 gram PoS *	Unigram and bigram of Part of Speech features.
	named entity recognition *	count of the present named entities in the sentence.
Syntactic features	parse tree depth [3, 12]	Indicates the depth of the sentence's parse tree.
	sub-clauses count [3, 12]	Indicates how many sub-clauses are in the sentence.
	verbal features *	counts of [modal, present, past, base form] verbs in the sentence
Discourse markers	keywords count [3, 12]	Number of existing keywords ('actually', 'because', etc.).
	numbers count *	Indicates how many numbers are there in the sentence.

In terms of lexical features, we found that *unigrams and bigrams of Part of Speech (PoS)* tags are very useful to capture the PoS patterns that are frequently

observed in argument components. Moreover, *named entity recognition* is a subtask of information retrieval that locates the named entities in unstructured text such as person names, organizations, quantities and time expressions. Such entities are usually used when stating a granted fact, reporting some incidents, or formulating a conclusion (i.e. in an argument component). Therefore, we take into account how many named entities appeared in the sentence as one feature to our model. We are using those two lexical features to improve the accuracy of the model and to make the identification of arguments more efficient and precise.

Furthermore, syntactic and grammatical features play an essential role for argument identification. In particular, the *depth of parse tree*, the *verbal features* and *count of sub-clauses* which clearly reflect the complexity of the sentence. This is important since evidence tend to appear in a complex sentence structure with more than one sub-clause. As far as we were able to find out in relevant literature [3] only the tense of the main verb of the sentence has been used to distinguish between claims and premises, however, the tense of the other verbs of the sentence is also helpful to make this identification more accurate. Indeed, sentences including several verbs in the past tense tend to be premises, whereas, the presence of many modal verbs and verbs in present tense makes the sentence more likely to be a claim.

Last but not least, we believe that the discourse markers present a direct indicator for argumentative text. For instance, the terms: 'consequently' and 'conclude that' are often followed by a claim, while the terms: 'for instance' and 'first of all', are mostly followed by a premise. Hence, we use a set of 286 discourse markers presented by A.Knott et al. [24] to generate the *keywords count* feature that reinforces argumentative text detection. Since statistics are generally used to support a claim, the existence of statistical numbers in a sentence (*numbers count* feature) makes it more likely to be identified as an argument. We chose to feed those features into an SVM model. This choice is justified by the fact that SVM performs effectively on small datasets and in high dimensional spaces.

3.4 Transfer Learning Model (DistilBERT- based)

Among many existent transformers, BERT [10] has recently gained a lot of attention. It seems to achieve the state of the art results in several NLP tasks [4–6, 25]. For our particular task, we performed different experiments using many BERT-based models (BERT base, RoBERTa-base[26], DistilRoBERTa , DistilBERT)[4] and achieved very similar results. Hence, we finally decided to use the DistilBERT given that it is 40% less than BERT in size with a relevant in-line performance and a faster training/testing time [8].

Fig. 1 describes the adopted pipeline to perform the text classification using DistilBERT. The first block is the Tokenizer that takes care of all the BERT input requirements: (1) It transforms the sentence's words into an array of DistilBERT tokens. (2) It adds the special starting token ([CLS] token). (3) It adds the necessary padding to have a unique size for all sentences (we set 128 as a

[4] We used Transformers from huggingface.co for our experiments.

Fig. 1. Transfer Learning Model Architecture

maximum length). The second block is the DistilBERT fine-tuned model, that outputs mainly a vector of length of 768 (default length). Our mission now is to adapt the output of this pre-trained model to our specific task. We achieve this by adding a third block, which is a linear layer applied on top of the DistilBERT transformer, and outputs a vector of size 2. The index of the maximum value in this vector represents the predicted class id. We trained the model for 3 epochs, using AdamW [27] as an optimizer and Cross Entropy for the loss calculation.

3.5 Overall Model (SVM + DistilBERT)

At this step, we have two models based on two completely different approaches. One is based on textual features while the other is based on the NLP transformer's ability of language understanding. Since they are two heterogeneous learning models, we chose to use the **stacking** ensemble method to combine their predictions.

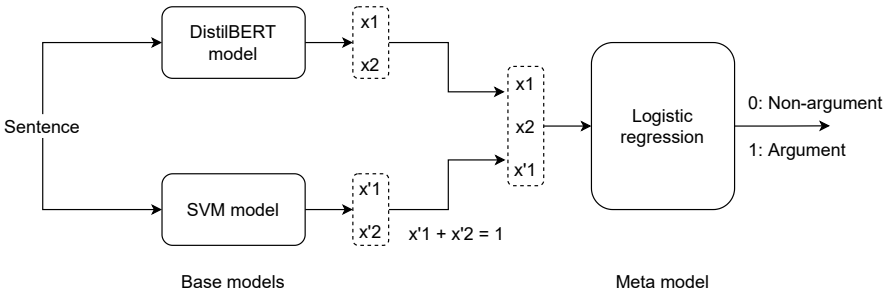

Fig. 2. Stacked Model Architecture

Fig. 2 presents the stacked model architecture, consisting of two main components: 1. the base models, that include the trained SVM model and the trained transformer based model (DistilBERT) in parallel, 2. the meta model, that will learn from the outputs of the two models to produce the final prediction of a sentence. In order to have an array of independent features for the meta-model, and since SVM outputs two probabilities $x'1$ and $x'2$ (i.e. $x'1 + x'2 = 1$), we consider only $x'1$. Whereas, $x1$ and $x2$ are two independent raw logits so both of them are considered. Given that we are dealing with a binary classification problem where the input features are independent, logistic regression serves well as a meta-model to accomplish the task. For the training/testing steps, we split

first the combined dataset into 75% training and 25% for the overall testing. This testing data remains unseen for all the models and it is used only for the final validation of the overall model. The base models are trained on the 75% training data. The training data of the meta model is prepared by 5-folds cross validation of the two base models. In each fold, the out-of-fold predictions are used as a part of the training data for the meta-model.

4 Evaluation

In this section, we discuss the performance of each of the individual learners apart and the final stacked model. In addition, we state some comparison with the most recent previous work[4] tackling the same problem of argument identification on the same datasets. We will further discuss the results shown in Table 2, 3 and Table 4.

In terms of SVM model, we can see that it works very well on the Essays corpus with an accuracy of 90.95% and F1-score of 83.75%. On the other hand, it seems less efficient on the Web Discourse corpus, where the transfer learning model provides better measurements. This can be interpreted by the formal structure of Student Essays compared to Web Discourse as we mentioned in Section 3.2. We conducted this evaluation in the context of a cross-domain argument identification, in the sense that we apply the model on different merged corpora. In these settings, SVM achieved an accuracy of 85.42%, using the textual features that learn a set of patterns in argument identification. In some cases[5], SVM fails to classify an argumentative sentence as an argument due to the absence of such necessary features. These limitations might be handled by understanding the meaning of the sentence using the characteristic of transfer learning through the pre-trained DistilBERT model. Of course, there are other cases where the contrary happens - SVM classifies correctly and DistilBert model fails. Hence, we have decided to combine these models.

Table 2. Achieved results on **Student Essays** corpus

Model	**Accuracy**	**Precision**	**Recall**	**F1-score**
SVM	0.9095	0.8730	0.8116	0.8375
DistilBERT	0.8727	0.8016	0.7477	0.7697
Stacking model	0.9162	0.8890	0.8195	0.8483

As we can see in the normalized confusion matrices (Fig. 3), SVM model reaches a higher percentage than DistilBERT-based model in terms of True Positive (TP) whereas the latter performs better than SVM for True Negative (TN).

[5] Here is an example (from Essays dataset) of an argument sentence that SVM fails to identify while DistilBERT succeeds: *"Personally, I think both government and common people should have the responsibility for the environment, but we need to analyze some specific situations."*

Therefore, the stacked model is getting the most out of both of them in terms of TN and TP, and thus it records a better classification accuracy, precision and recall as shown in Table 4.

Table 3. Achieved results on **Web Discourse** corpus

Model	Accuracy	Precision	Recall	F1-score
SVM	0.7437	0.7051	0.5882	0.5874
DistilBERT	0.7799	0.7718	0.6484	0.6655
Stacking model	0.7855	0.7449	0.6958	0.7113

In recent work [4], the authors implemented a BERT-based transfer model on different corpora including the two datasets we use in this paper. Our stacked model overcomes theirs on the Student Essays achieving an accuracy of 91.62% and F1-score of 84.83% compared to their accuracy of 80.00% and F1-score of 85.19%. On the Web Discourse corpus, we have similar accuracy values(78.5% to 80.00%) while on the level of the combined model, our approach achieved better performance even though they have investigated on more training corpora.

Table 4. Achieved Results on the Merged Corpora (**Student Essays and Web Discourse**)

Model	Accuracy	Precision	Recall	F1-score
SVM	0.8542	0.8037	0.7012	0.7331
DistilBERT	0.8587	0.7887	0.7529	0.7683
Stacking model	0.8780	0.8326	0.7659	0.7921

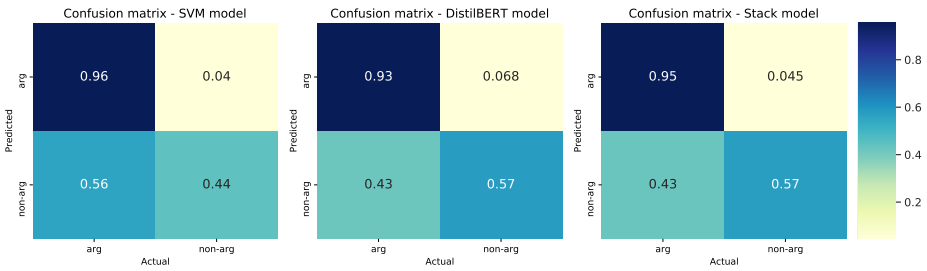

Fig. 3. Normalized Confusion Matrices

Furthermore, we emphasize that the idea of combining two different approaches is not only about the improvement of results, but also a step forward for the model's interpretability. On one hand, deep learning models(transformers in our case) reduce the task of feature engineering. Yet, it is difficult for humans to fully understand their behaviour. On the other hand, the direct feature en-

gineering involved in classical ML makes those models more interpretable and understandable.

5 Conclusion

In this paper, we present a novel approach in the field of argument mining. Our model leverages the ensemble learning stacking method to combine a classical machine learning model and a deep transfer learning model to benefit from the advantages of both. The evaluation of the presented model shows good performance in the task of argument identification with an accuracy of 0.8780 and F1-score of 0.7921 on the two merged corpora. The proposed approach is a first insight into combining different argument mining techniques and methods to build a more general and robust model. In future work, we consider to extend our approach to other corpora from different domains. We plan also to use the model stacking method to cover additional argument mining tasks such as argument components classification and argument structure identification.

Acknowledgements

SPONSORED BY THE

MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR, DE LA RECHERCHE ET DE L'INNOVATION
Liberté
Égalité
Fraternité

Federal Ministry of Education and Research

This work was supported by the French Ministry of Higher Education and Research. It has been also co-funded by the German Federal Ministry of Education and Research (BMBF) under the funding code 01—S20049.

References

1. Trudy Govier. A practical study of argument, (belmont. *CA: Wadsworth*, 2001.
2. Connie A Missimer. *Good arguments: An introduction to critical thinking*. Prentice Hall, 1995.
3. Christian Stab and Iryna Gurevych. Identifying argumentative discourse structures in persuasive essays. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 46–56, 2014.
4. Thiemo Wambsganss, Nikolaos Molyndris, and Matthias Söllner. Unlocking transfer learning in argumentation mining: A domain-independent modelling approach. In *15th International Conference on Wirtschaftsinformatik*, 2020.
5. Nils Reimers, Benjamin Schiller, Tilman Beck, Johannes Daxenberger, Christian Stab, and Iryna Gurevych. Classification and clustering of arguments with contextualized word embeddings. *arXiv preprint arXiv:1906.09821*, 2019.
6. Timothy Niven and Hung-Yu Kao. Probing neural network comprehension of natural language arguments. *arXiv preprint arXiv:1907.07355*, 2019.
7. Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.
8. Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

9. Omer Sagi and Lior Rokach. Ensemble learning: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(4):e1249, 2018.
10. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
11. Thomas Wolf, Julien Chaumond, Lysandre Debut, Victor Sanh, Clement Delangue, Anthony Moi, Pierric Cistac, Morgan Funtowicz, Joe Davison, Sam Shleifer, et al. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, 2020.
12. Marie-Francine Moens, Erik Boiy, Raquel Mochales Palau, and Chris Reed. Automatic detection of arguments in legal texts. In *Proceedings of the 11th international conference on Artificial intelligence and law*, pages 225–230, 2007.
13. Raquel Mochales Palau and Marie-Francine Moens. Argumentation mining: the detection, classification and structure of arguments in text. In *Proceedings of the 12th international conference on artificial intelligence and law*, pages 98–107, 2009.
14. Christian Stab and Iryna Gurevych. Annotating argument components and relations in persuasive essays. In *Proceedings of COLING 2014, the 25th international conference on computational linguistics: Technical papers*, pages 1501–1510, 2014.
15. Christian Stab and Iryna Gurevych. Parsing argumentation structures in persuasive essays. *Computational Linguistics*, 43(3):619–659, 2017.
16. Ivan Habernal and Iryna Gurevych. Argumentation mining in user-generated web discourse. *Computational Linguistics*, 43(1):125–179, 2017.
17. Johannes Daxenberger, Steffen Eger, Ivan Habernal, Christian Stab, and Iryna Gurevych. What is the essence of a claim? cross-domain claim identification. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2055–2066, Copenhagen, Denmark, September 2017. Association for Computational Linguistics.
18. Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.
19. Davide Liga and Monica Palmirani. Transfer learning with sentence embeddings for argumentative evidence classification. 2020.
20. Mark J Van der Laan, Eric C Polley, and Alan E Hubbard. Super learner. 2007.
21. Régis Goubin, Dorian Lefeuvre, Alaa Alhamzeh, Jelena Mitrovic, Elöd Egyed-Zsigmond, and Leopold Ghemmogne Fossi. Bots and gender profiling using a multi-layer architecture. In *CLEF (Working Notes)*, 2019.
22. Giovanni Ciccone, Arthur Sultan, Léa Laporte, Elod Egyed-Zsigmond, Alaa Alhamzeh, and Michael Granitzer. Stacked gender prediction from tweet texts and images notebook for pan at clef 2018. In *CLEF 2018-Conference and Labs of the Evaluation*, page 11p, 2018.
23. Stephen E Toulmin. *The uses of argument*. Cambridge university press, 2003.
24. Alistair Knott and Robert Dale. Using linguistic phenomena to motivate a set of rhetorical relations. 08 1997.
25. Tommaso Caselli, Valerio Basile, Jelena Mitrović, Inga Kartoziya, and Michael Granitzer. I Feel Offended, Don't Be Abusive! Implicit/Explicit Messages in Offensive and Abusive Language. In *Proceedings of LREC*, 2020.
26. Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.
27. Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.