Empirical Study of the Model Generalization for Argument Mining in Cross-Domain and Cross-Topic Settings

Alaa Alhamzeh[1,2(✉)], Előd Egyed-Zsigmond[1], Dorra El Mekki[2], Abderrazzak El Khayari[2], Jelena Mitrović[2,3], Lionel Brunie[1], and Harald Kosch[2]

[1] INSA de Lyon, 20 Avenue Albert Einstein, 69100 Villeurbanne, France
{Elod.Egyed-zsigmond,Lionel.Brunie}@insa-lyon.fr
[2] Universität Passau, Innstraße 41, 94032 Passau, Germany
{Alaa.Alhamzeh,Jelena.Mitrovic,Harald.Kosch}@uni-passau.de,
{elmekk01,elkhay01}@ads.uni-passau.de
[3] Institute for AI Research and Development, Fruškogorska 1, 21000 Novi Sad, Serbia

Abstract. To date, the number of studies that address the generalization of argument models is still relatively small. In this study, we extend our stacking model from argument identification to an argument unit classification task. Using this model, and for each of the learned tasks, we address three real-world scenarios concerning the model robustness over multiple datasets, different domains and topics. Consequently, we first compare single-datset learning (SDL) with multi-dataset learning (MDL). Second, we examine the model generalization over completely unseen dataset in our cross-domain experiments. Third, we study the effect of sample and topic sizes on the model performance in our cross-topic experiments. We conclude that, in most cases, the ensemble learning stacking approach is more stable over the generalization tests than a transfer learning DistilBERT model. In addition, the argument identification task seems to be easier to generalize across shifted domains than argument unit classification. This work aims at filling the gap between computational argumentation and applied machine learning with regard to the model generalization.

Keywords: Argument mining · Robustness · Generalization · Multi-dataset learning · Cross-domain · Cross-topic

1 Introduction

Human communication is a complex function of language, facial expressions, tone of speech, and body language. We communicate to express our feelings, opinions, and beliefs which are a result of many arguments we believe in. Those

A. Hameurlain and A. M. Tjoa (Eds.): *Transactions on Large-Scale Data- and Knowledge-Centered Systems LII*, LNCS 13470, pp. 103–126, 2022.
https://doi.org/10.1007/978-3-662-66146-8_5

arguments may have been influenced by a life experience, a known fact, analogies or something else. However, to persuade the other party by our point of view, we have to present and explain those different arguments to them. Argumentation, therefore, has been remarked and studied since the 6th century B.C. by Ancient Greek philosophers. Aristotle's Logic [1], and in particular his Theory of the syllogism has laid the groundwork for the definitions of logical reasoning and arguments of today.

Given this historical dimension of argumentation studies, there has been a large number of different argumentation theories which align with different aspects of argumentation [2]. The same applies to the argument quality criteria. Similarly, there have been numerous proposals for assessing the merits or defects of an argument.

The argument, in its simplest form, consists of one claim (also called a conclusion [3]) supported by at least one premise [4]. The premise, which is also known as the evidence, is the reason or justification for the connected claim. Figure 1 shows an example of an argumentative student essay [5] regarding the controversial topic "having children and quality of life". In this example, the argument units 1, 2, and 3 are premises related to the claim marked in argument unit 4. Moreover, we see in this example an instance for a "Support" relation between premise 1, premise 3 and the final claim as well as to an "Attack" relation between the premise 2 and the final claim. We have to note at this point, that an attack relation does not necessarily imply a bad or invalid argument. Indeed, discussing a potential rebuttal is a common strategy to prevent any potential criticism. In other words, the author states the contrary opinion (argument unit 2) and then states why it is not relevant (argument unit 3) which makes the overall argument stronger and more likely to be convincing.

"[Raising your own child is like having an important goal in your live]$_1$. Admittedly, [you will have great responsibilities and you also will have sleepiness nights]$_2$ but [these drawbacks will turn into a valuable experience when your kids become older]$_3$. Therefore, [Having children is the ultimate bliss in our lives]$_4$."

Fig. 1. Example of arguments (as taken from Student Essays Corpus [5])

Argument mining (AM) has been a self-established field of natural language processing in the last decade. That can be justified by the wide spectrum of its practical applications. For instance, computer-assisted writing [6], fact checking [7], and decision support systems like the one presented in [3] for legal counselling and [8] for comparative question answering. That points out the value of argumentation in interdisciplinary settings.

However, generalizing over different domains is still one of the hardest challenges in computational argumentation. Moreover, there is still a research space for improving the assessment of model stability from machine learning point of

view. The model robustness is its performance stability over new variants comparing to the training stage. Mainly, robustness over new data distribution (e.g., [9]) or different model runs (e.g., [10]). We examine both aspects in our experiments. Therefore, we aim in this study, to extend our previous work [11], by performing a deep analysis on our stacked model from two perspectives:

- Machine learning perspective: we evaluate the robustness of our model in different cross-domain and cross-topic settings.
- Argument mining perspective: we perform a comprehensive feature analysis on how "argumentativeness" is similarly (mis)captured across diverse datasets.

To this end, we also integrate a third corpus and apply multiple methodologies for two argumentation classification tasks. Consequently, the contributions of this study are:

1. First, we move a step towards another argumentation task which is argument unit classification task (premise/claim classification). We investigate that using the same two primarily used corpora, with a new integrated dataset from the IBM Project Debater[1]. We, henceforth, apply all of our experiments, for both learned tasks: argument identification and argument unit classification.
2. We start by a preliminary step with respect to the model selection on our stacked approach, and we report on the best combination of features to be adopted in our further experiments.
3. We examine if including more data during the training stage (Multi-dataset learning), would increase the performance reported using one dataset (Single-dataset learning).
4. Moreover, to assess the model generalization ability and robustness over shifted data distributions, we set up a cross-domain experiment where we test the model on a completely unseen corpus.
5. Similarly, we apply cross-topic experiments, where, in each run, a unique set of unseen topics is saved apart for testing. In addition, this experiment aims at examining how the number of training topics influences the model generalization performance over unseen topics.
6. To foster the work in this area, our source-code is available publicly through our github repository[2]

This paper is organized as follows: in Sect. 2, we take a close look at the conceptual background of our work as well as the literature studies considering our points of interest. In Sect. 3, we present the problem statement as well as an overview on the data, and our stacked models that compose the cornerstone of the experiments examined in this paper. We employ a deep feature analysis and model selection and we examine the generalization of the selected model on various cross scenarios in Sect. 4. Finally, we discuss the overall research questions and future work in Sect. 5.

[1] https://research.ibm.com/interactive/project-debater/index.html.
[2] https://github.com/Alaa-Ah/Stacked-Model-for-Argument-Mining.

2 Related Work

The basic challenge of the AM research field is its variance over domains and topics. The model falls short in shifted domain settings. Therefore, the search for a domain-agnostic model was a point of interest for many researches. One of the proposed solutions towards this, is to use transfer learning models. Liga et al. [12] aimed at discriminating evidence related to argumentation schemes. They used three different pre-trained transformers to generate multiple sentence embeddings, then trained classifiers on it. Wambsganss et al. [13] proposed an approach for argument identification using BERT on multiple datasets. Our stacked model overcomes theirs on the Student Essays corpus 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.

Besides transformers, adversarial learning has also been selected to test the model robustness over shifted data distribution by providing deceptive input. Indeed, the assumption behind adversarial learning in NLP, is that by generating variant samples across several domains, deep networks are resistant not only to heterogeneous texts but also to linguistic bias and noise [14]. Tzeng et al. proposed in [15] a novel unsupervised domain adaptation framework to identify transferable features that are suitable for two different domains. This approach showed promising results in unsupervised tasks.

Over time, cross-domain AM became a must. However, it has been mainly studied in a multi-dataset manner (or as multi-dataset learning (MDL) as addressed by [16]). For example, in the work of Ajjour et al. [17], they extend the argument unit segmentation task to investigate the robustness of the model while testing on three different corpora; the essays corpus [5], the editorials corpus [18], and the web discourse corpus [19]. Their proposed argument unit segmentation system is based on a neural network model incorporating features on a word-level setting on the in-domain level as well as cross-domain level. Their results show that structural and semantic features are the most effective in segmenting argument units across domains, whereas semantic features are best at identifying the boundaries of argumentative units within one domain. However, in their study, features are extracted at the token level whereas in our approach, we tackle the sentence level classification for our experiments within and across domains and for both argumentative sentence detection and argument component classification tasks.

In addition, [20] proposed the ArguWeb, a cross-domain argument mining CNN-based framework designed to first extract the argument from the web then segment it and classify its units. This approach tackles the two subtasks of argument mining: argument detection and argument component classification on both in-domain and cross-domain scenarios. In terms of cross-domain, the model was trained on two corpora and tested each time on a third one. The evaluation of the model's performance was conducted using character-level CNN, word-based CNN, SVM, and Naive Bayes in two scenarios: in-domain and cross-domain.

The results show that the character-level CNN outperforms other models when testing on web-extracted data such as the web-discourse corpus.

With respect to machine learning, we usually evaluate the prediction of models on an unseen split of the dataset and use that to report the performance of the model. Yet, this generalization could be limited to data that follow the same distribution which the model has already been trained on. In other words, the model memorized it rather that generalized over it. This issue has been studied by [21] and they conclude that quantifying train/test overlap is crucial to assessing real world applicability of machine learning in NLP tasks, especially when the training data is not large enough such as in a shared task case. According to [22], the key issue is that the algorithm training error provides an optimistically biased estimation, especially when the number of training samples is small. Therefore, many methods have been suggested to prevent this deviation from the empirical measurements. [10] investigated whether the linguistic generalization behaviour of a given neural architecture is consistent across multiple instances (i.e., runs) of that architecture. They found that models that differ only in their initial weights and the order of training examples can vary substantially in out-of-distribution linguistic generalization. Therefore, we always consider the average of 5 different runs along all our paper experiments.

Another interesting approach to measure robustness of a model is by using compositional generalization, which combines different parts of the test samples. An example would be, in image classification, adding rain or snow effect or rotating images in order to see if a self-driving car still recognizes the stop sign. This out-of-distribution generalization is known as Domain Generalization, which aims to learn a model from one or several different but related domains (i.e., diverse training datasets) that will generalize well on unseen testing domains [23]. We adopt this definition in our cross-domain experiments. We also derive a similar strategy with respect to unseen topics from the same dataset in our cross-topic experiments.

We also want to point out that several research fields are closely related to domain generalization, including but not limited to: transfer learning, multi-task learning, ensemble learning and zero-shot learning [23]. In our models, we have combined ensemble learning using a stacking approach and transfer learning using DistilBERT.

However, before we extend upon the experiments, we apply deep feature analysis and model selection, motivated by the work of [24] to set up the best model configuration for each of the two addressed AM tasks, namely argument identification and argument unit classification. To this end, we used three different corpora which are highly investigated in AM studies.

3 Method

In this section, we present the problem statement, the used corpora, and an overview on our stacking model [11], which we will further use in all our experiments.

3.1 Problem Statement

The problem of argument mining is vast and can be seen as a set of several sub-tasks. In this paper, we consider two of them: argument identification and argument unit classification as discourse analysis problems. In our approach, we classify the text at sentence level since it is less common to have two parts of an argument in one sentence. For example, Stab et al. [25] reported, for Student Essays corpus, that only 8.2% of all sentences need to be split in order to identify argument components. Moreover, a sentence with two different argumentative components will still be valid (and considered as an argument) on the level of argument identification task. Therefore, the first step will be always to apply a sentence segmentation then a binary classification of each sentence with respect to the particular task. In addition to the argument mining tasks, we aim at answering the following questions:

(1) What is the minimal set of features that can capture arguments and their units over different datasets? (Sect. 4.1).
(2) Is it beneficial to include data from different argument models to increase accuracy? (Sect. 4.2).
(3) To what extent is our AM approach independent of domain and data diversity? To tackle this point, we run a cross-domain experiment where we test on a completely unseen corpus (Sect. 4.3), and cross-topic experiments where we test on unseen topics from the same corpus (Sect. 4.4).

3.2 Data Description

In our work, we use three publicly available corpora:

The **Student Essays corpus**: contains 402 Essays about various controversial topics. This data has been introduced by Stab et al. [5]. The annotation covers three argument components, namely, 'major claim', 'claim', and 'premise'. Moreover, it presents the support/attack relations between them. Hence, it was used in several argument mining tasks. The dataset also includes one file called 'prompts' which describes the question behind each essay. We consider this 'prompt' as the topic of the essay.

The **User-generated Web Discourse corpus** is a smaller dataset that contains 340 documents about 6 controversial topics in education such as mainstreaming and prayer in schools. 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 [19] following Toulmin's model [4]. Thus, it covers the argument components: 'claim', 'premise', 'backing' and 'rebuttal'.

The **IBM corpus** [26] consists of 2683 manually annotated argument components derived from Wikipedia articles on 33 controversial topics. It contains 1392 labeled claims and 1291 labeled evidence for 350 distinct claims in 12 different topics. In other words, there are only 1291 evidences derived from only

Table 1. Class distributions for all used datasets

Dataset	#Premise	#Claim	#Non-arg	#Topics
StudentEssays	3510	1949	1358	372
WebDiscourse	830	195	411	6
IBM	1291	1392	0	33

12 topics, while there are 1042 claims unsupported by evidence derived from 21 different topics. This dataset does not include a "Non-argument" label so we could not use it for the argument identification task. Instead, we used it only for experiments on argument unit classification.

Table 1 shows the class distributions for the three datasets. Moreover, different samples of those datasets are expressed in Table 2. We can clearly observe that they do not share the same characteristic like the text length and organization. This makes it more challenging to design a model that generalizes well over them.

Table 2. Text examples from the different datasets

Student Essays	IBM article	Web Discourse
"First of all, through cooperation, children can learn about interpersonal skills which are significant in the future life of all students. What we acquired from team work is not only how to achieve the same goal with others but more importantly, how to get along with others. On the other hand, the significance of competition is that how to become more excellence to gain the victory. Hence it is always said that competition makes the society more effective."	"Exposure to violent video games causes at least a temporary increase in aggression and this exposure correlates with aggression in the real world. The most recent large scale meta-analysis– examining *130 studies* with over *130,000 subjects* worldwide– concluded that exposure to violent video games causes both short term and long term aggression in players."	"I think it is a very loving thing, a good and decent thing to send children to a private school!"

3.3 Models of Our Ensemble Learning Approach

The main assumption behind ensemble learning 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 [27]. We adopted this learning approach in our model in order to reach a trade-off between the pros and cons

of classical and deep learning models. It is well known that the data scale drives deep learning progress, yet data labeling is an expensive and time-consuming process, especially given the nature of fine-grained hierarchical argumentation schemes as in the student essays corpus [24]. Whereas, in a small training set regime (which is usually the case in AM available datasets), a simple classical machine learning model may outperform a more complex neural network. In addition, training a classical model and interpreting it is much faster and easier than a neural network based one. On the other hand, traditional machine learning algorithms fall short as soon as the testing data distribution or the target task are not the same as the distribution of the training data and the learned task. In contrast, in transfer learning, the learning of new tasks relies on previously learned ones. The algorithm can store and access knowledge over different corpora and different tasks. Hence, we studied both directions of the learning algorithms and composed our final model as follows:

Classical Machine Learning Model - SVM
In terms of the first base model, we defined a set of diverse features inspired by the works of [24,28]. We organize those features in sets of structural, lexical, and syntactic features in addition to discourse markers. Table 3 shows a complete list of each group of features as well as their description (more details about the reasoning behind each one can be found in our previous work [11]).

Transfer Learning Model (DistilBERT- Based)
Transfer learning 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 share some similarities. Recently, BERT-like models got a lot of attention since they achieve state of the art results in different NLP tasks.

DistilBert [30] is trained on the same corpus like BERT [31] which consists of 16GB data (approximately 3.3 Billion words) from books corpus and Wikipedia. This large corpus of diverse topics enables it to show robust performance for domain-shift problems [32]. Figure 2 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. 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 [33] as an optimizer and cross entropy for the loss calculation.

Table 3. Textual features. Our original added features are marked with '*'

Group	Feature	Description
Structural	sentence position [29]	Indicates the index of the sentence in the document.
	tokens count [28,29]	Indicates the count of tokens (words) in the sentence.
	question mark ending [29]	Boolean feature.
	punctuation marks count [29]	Indicates the number of punctuation marks in the sentence.
Lexical	1-3 g bow [28,29]	Unigrams, bigrams and trigram BoW features.
	1-2 g POS *	Unigram and bigram of POS features.
	NER *	count of the present named entities in the sentence.
Syntactic	parse tree depth [28,29]	Indicates the depth of the sentence's parse tree.
	sub-clauses count [28,29]	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 [28,29]	Number of existing argument indicators ('actually','because', etc.).
	numbers count *	Indicates how many numbers in the sentence

Fig. 2. Transfer learning model architecture using DistilBERT

Ensemble Learning - Stacking (SVM + DistilBERT)
At this step, we have two heterogeneous based learners. One is based on textual features while the other is based on the NLP transformer's ability of language understanding. Therefore, a stacking approach fits perfectly to combine their predictions. As shown in Fig. 3, the outputs of SVM and DistilBert are used as input to the meta model that will learn how to produce the final prediction of a sentence based on the outputs of the base models. In order to have an array of independent features for the meta-model, and since SVM produces 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

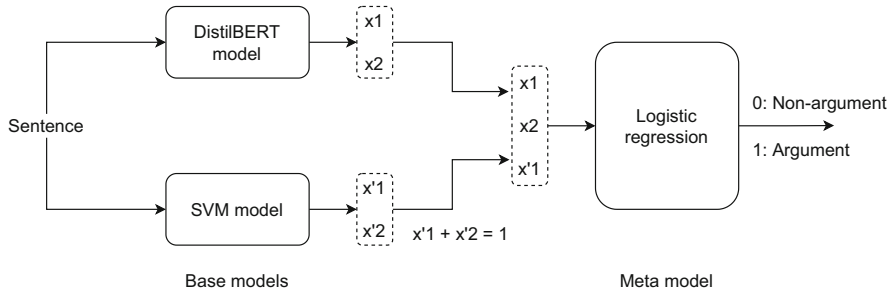

Fig. 3. Stacked model architecture for argument identification task

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 Experiments

In this section, we first take a preliminary stage to perform a model selection on our stacking method in Sect. 4.1. With the best found configurations, we move to our contribution regarding the examination of the model robustness and domain generalization in Sects. 4.2, 4.3 and 4.4. In all of our experiments, we address both argument identification and argument unit classification tasks.

4.1 Feature Analysis and Model Selection

We revisited our feature engineering part to assure the impact of each feature on its own and in correlation (or dependency) with other features. Including more features in the training can be problematic since it can increase space and computational time complexity. It can also introduce some noise according to unexpected value changes. These shortcomings are known as the curse of dimensionality. The main solution for dimensionality reduction is feature selection where different methods can be applied. In our work, we have first applied a *filter method* that is based on variance threshold such that we can figure out any features that do not vary widely between the three classes. We achieve that by visualizing the distribution's histogram of each feature. This helps us to see if a feature is important and improves the performance or if it has a redundant effect (or even no effect) on the final output. We present two examples in the following:

Figure 4 suggests that sentence position in the input paragraph correlates positively with premise sentences. In particular, with the positions 1 to 5. This means that a sentence that is stated earlier in the paragraph is more likely to be a premise than a claim or non-argument. We can see also that the value

of position zero is very frequent since in WD and IBM, we do not have long paragraphs like in SE, rather it may be only one sentence.

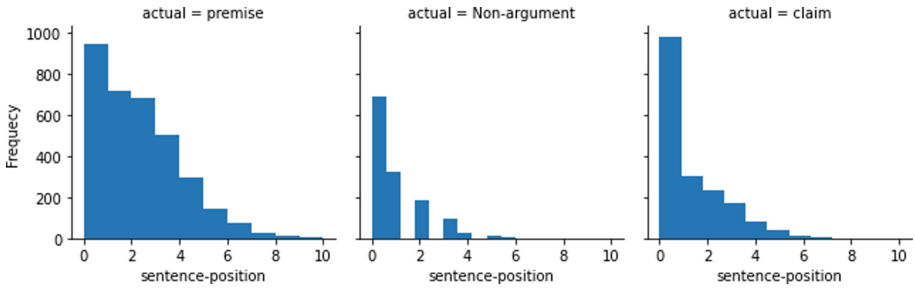

Fig. 4. Histograms of the *sentence position* feature

Similarly, Fig. 5 reflects the distribution of the punctuation marks over the three classes. We obviously can see that non-argument text tends to have more punctuation marks than argumentative text. Also, in terms of premise/claim classification, sentences with more than seven punctuation marks are only premises.

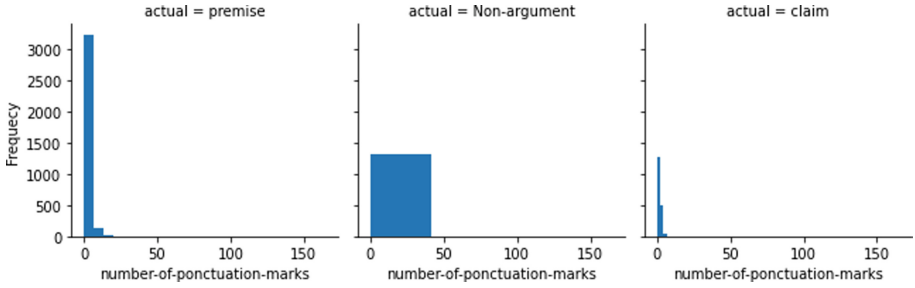

Fig. 5. Histograms of *number of punctuation marks* feature

Both "sentence position" and "number of punctuation marks" are part of our structural features which proved to be very essential in our model selection process. We identify the best performing model by conducting a *feature ablation tests*. Consequently, in order to determine the best configuration for our stacked model, we apply at this step a kind of a *wrapper method* that iterates through different combinations of features and performs a model retrain on each. For this model assessment, we adopt the accuracy as well as the weighted average metrics of precision, recall and F1-score since our data is imbalanced. The feature combination which results in the best model performance metrics for each AM task is selected.

Since the effect of different groups of features will be on the SVM performance in the first place, and subsequently on the stacking model that combines SVM with DistilBERT predictions, we report in this section, both SVM and stacked model results for the different settings. Moreover, in order to ensure more statistically significant testing, we have conducted for every set of features 5 runs over 5 different seeds, and internally 5-fold cross validation. That means for each set of features the model is tested 25 times. We report the weighted mean and the standard deviation of those runs for each classification task.

Model Selection on Argument Identification Task

Table 4 shows the results of argument identification task using SVM over different groups of features. Our findings suggest that SVM scores the best performance using lexical, structural and syntactical features with a slightly better weighted F1-score of 85.7% than SVM with all features or with lexical, structural and discourse markers (W-F1 score = 85.6%) while they all achieve the same accuracy of 86.1%

Table 4. Results of feature analysis on argument identification task using SVM on SE and WD

	W-Precision		**W-Recall**		**W-F1 score**		**Accuracy**	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std
lexical	0.782	±0.001	0.807	±0.001	0.794	±0.001	0.807	±0.001
structural	0.825	±0.0	0.838	±0.0	0.831	±0.0	0.838	±0.0
syntactic	0.617	±0.0	0.786	±0.0	0.691	±0.0	0.786	±0.0
discourse markers	0.617	±0.0	0.786	±0.0	0.691	±0.0	0.786	±0.0
lexical, structural	*0.849*	±0.001	*0.858*	±0.001	*0.853*	±0.001	*0.858*	±0.001
lexical, structural, syntactical	**0.853**	±0.001	**0.861**	±0.001	**0.857**	±0.0	**0.861**	±0.001
lexical, structural, discourse markers	*0.852*	±0.0	*0.861*	±0.0	*0.856*	±0.0	*0.861*	±0.0
all features	*0.852*	±0.0	*0.861*	±0.0	*0.856*	±0.001	*0.861*	±0.0

Similarly, Table 5 confirms that the combination of structural, lexical and syntactical features achieves the best performance at the level of the stacked model. However, we observe that the scored mean of different settings is similar, especially when considering the structural features. According to the student-t test [34], when structural features are considered, the p-value exceeds 5%. Hence, we cannot claim that including (excluding) some features, except for structural and lexical, makes a huge difference on our model. Yet, we adopt the best performing model which empirically proved to be the model with structural, lexical and syntactical features for argument identification task. We, henceforth, use these settings for the upcoming experiments on this particular task.

Table 5. Results of model selection on argument identification using Stacked model on SE and WD

	W-Precision		**W-Recall**		**W-F1 score**		**Accuracy**	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std
lexical	0.830	±0.004	0.842	±0.003	0.836	±0.003	0.842	±0.003
structural	0.851	±0.006	0.86	±0.005	0.856	±0.006	0.860	±0.005
syntactic	0.831	±0.006	0.843	±0.005	0.837	±0.006	0.843	±0.005
discourse markers	0.831	±0.007	0.843	±0.006	0.837	±0.007	0.843	±0.006
lexical, structural	*0.862*	±0.002	*0.869*	±0.002	*0.866*	±0.001	*0.869*	±0.002
lexical, structural, syntactical	**0.863**	±0.003	**0.870**	±0.003	**0.866**	±0.003	**0.870**	±0.003
lexical, structural, discourse markers	*0.861*	±0.004	*0.868*	±0.004	*0.865*	±0.004	*0.868*	±0.004
all features	*0.861*	±0.003	*0.868*	±0.002	*0.865*	±0.003	*0.868*	±0.002

Model Selection on Argument Unit Classification

To train the model on argument unit classification (i.e., premise/claim classification), we transform the feature "*Keywords count*" that indicates the count of any argument indicator, to two features: "*premise-indicators-count*" and "*claim-indicators-count*".

Furthermore, we also integrate a new dataset: IBM (cf. Table 1) and we further employ the model selection experiments as in the previous AM task. Table 6 confirms that SVM with all features delivers slightly better results compared to the other sub-combinations of features.

In terms of the stacked model, beside the semantic conceptual features that DistilBERT learns, we observe that the structural features are the most dominant proprieties that help to discriminate premises from claims in the three used corpora. However, they achieve a slight difference in comparison to their combination with lexical features and to the all features performance, as shown in Table 7. This finding is similar to the one by [17], which show that structural and semantic features are the most effective in segmenting argument units across domains.

Furthermore, Figs. 6 and 7 reports the F1 scores, and standard deviation, for SVM, DistilBERT and the stacked model across the different sets of features. We can observe that the stacked model scores at least the same performance of DistilBERT, and it improves over once the SVM classifier obtains a minimum score of 80% which is verified in most cases.

To sum up, in Sect. 4.1, we applied an in-depth feature analysis and model selection in two-folds: argument identification and argument unit classification. According to our findings, we ignore, henceforth, the features that lead to minor short-term wins, and we keep only the structural features for argument unit classification, and structural, lexical and syntactical features for argument identification task.

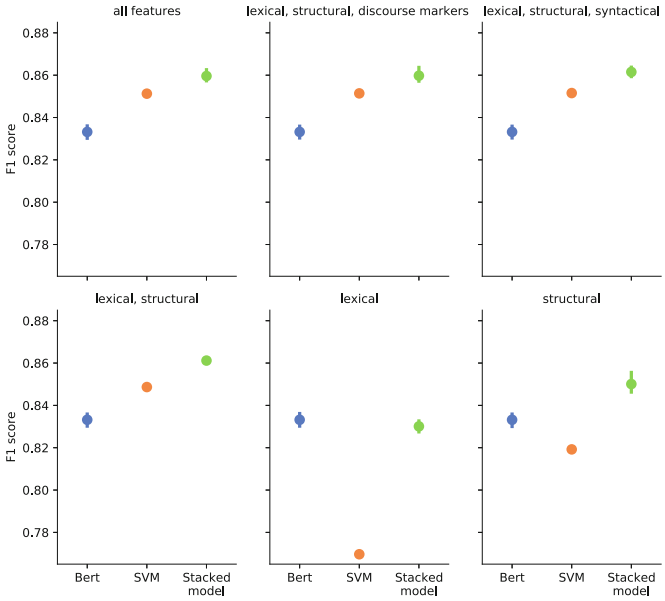

Fig. 6. Effect of feature selection on the argument identification task

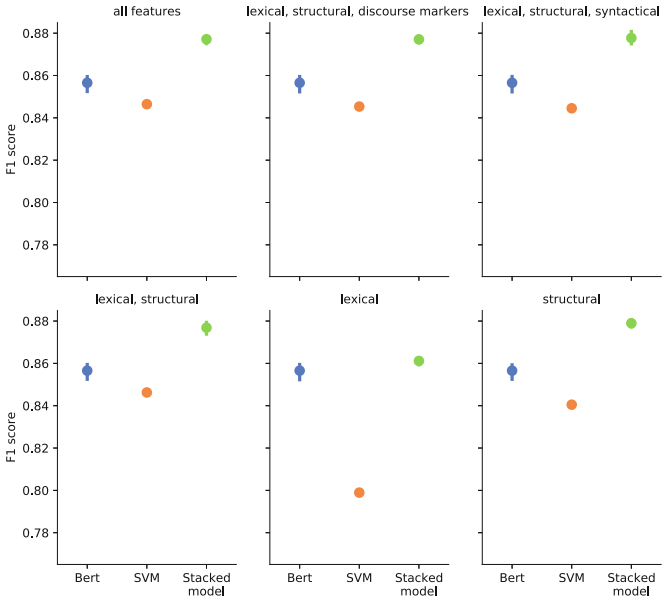

Fig. 7. Effect of feature selection on argument unit classification task

Table 6. Results of feature analysis on argument unit classification task using SVM on SE, WD and IBM datasets

	W-Precision		**W-Recall**		**W-F1 score**		**Accuracy**	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std
lexical	0.802	±0.001	0.803	±0.001	0.802	±0.001	0.803	±0.001
structural	*0.840*	±0.0	*0.841*	±0.0	*0.841*	±0.0	*0.841*	±0.0
syntactic	0.378	±0.0	0.615	±0.0	0.468	±0.0	0.615	±0.0
discourse markers	0.633	±0.0	0.648	±0.0	0.640	±0.0	0.648	±0.0
lexical, structural	*0.847*	±0.001	*0.848*	±0.001	*0.847*	±0.001	*0.848*	±0.001
lexical, structural, syntactical	*0.846*	±0.0	*0.846*	±0.0	*0.846*	±0.001	*0.846*	±0.0
lexical, structural, discourse markers	*0.846*	±0.0	*0.847*	±0.0	*0.847*	±0.001	*0.847*	±0.0
all features	**0.848**	±0.0	**0.848**	±0.0	**0.848**	±0.0	**0.848**	±0.0

Table 7. Results of model selection on argument unit classification task using Stacked model on SE, WD, and IBM datasets

	W-Precision		**W-Recall**		**W-F1 score**		**Accuracy**	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std
lexical	0.862	±0.002	0.863	±0.002	0.862	±0.002	0.863	±0.002
structural	**0.88**	±0.002	**0.88**	±0.002	**0.88**	±0.002	**0.88**	±0.002
syntactic	0.857	±0.003	0.857	±0.002	0.857	±0.002	0.857	±0.002
discourse markers	0.858	±0.003	0.858	±0.003	0.858	±0.002	0.858	±0.003
lexical, structural	*0.878*	±0.003	*0.878*	±0.003	*0.878*	±0.003	*0.878*	±0.003
lexical, structural, syntactical	*0.878*	±0.004	*0.879*	±0.004	*0.879*	±0.004	*0.879*	±0.004
lexical, structural, discourse markers	*0.878*	±0.002	*0.878*	±0.002	*0.878*	±0.002	*0.878*	±0.002
all features	*0.878*	±0.002	*0.878*	±0.002	*0.878*	±0.003	*0.878*	±0.002

4.2 Single-Dataset Learning (SDL) Vs. Multi-dataset Learning (MDL)

This experiment is intended to determine whether incorporating more datasets in the training step will generate a significant, positive impact on the robustness of the stacked model with respect to the test data, taking into account that our available datasets are relatively small. Consequently, we compare the outcomes of single-dataset learning and multi-dataset learning approaches.

In the SDL setup, we train and test the model on each dataset individually while in the MDL setup, we train the model on all datasets, but test on individual test split (20%) of the particular dataset. This methodology allows us to report performance scores on each dataset separately while training our model on a single versus multiple datasets.

We examine our model in these settings for the two trained tasks; argument identification and argument unit classification. However, since IBM has only the labels of argument components, we run the argument identification experiments

using WD and SE datasets, whereas we use WD, SE, and IBM for the argument unit classification experiments. We use for each task its best stacked model configuration conducted in Sect. 4.1.

Table 8. SDL vs. MDL argument identification using the stacked model.

		W-Precision		**W-Recall**		**W-F1 score**		**Accuracy**	
	Dataset	Mean	Std	Mean	Std	Mean	Std	Mean	Std
SDL	SE	**0.918**	±0.002	**0.92**	±0.002	**0.919**	±0.002	**0.92**	±0.002
	WD	**0.771**	±0.014	**0.776**	±0.011	**0.773**	±0.014	**0.776**	±0.011
MDL	SE	0.877	±0.006	0.881	±0.004	0.879	±0.004	0.881	±0.004
	WD	0.749	±0.011	0.765	±0.009	0.757	±0.015	0.765	±0.009

According to Table 8 and Table 9, we observe an expected drop in the performance for all datasets between the SDL and MDL setups. Yet, our stacked model is still able, in all the cases, to produce reliable accuracy and F1-score. Nevertheless, detecting argumentative text proved to be an intrinsically more generalized task than determining the premises and claims. For example, the variation of F1-score between the two settings, is in the range of $[-2\%, -4\%]$ for argument identification, while it moves to the range of $[-7\%, -9\%]$ for argument unit classification task.

These evaluation results also suggest that a single learning is always better when we are sure that our future targeted data follows the same or a very close distribution to the training one. This allows for better capturing of the dataset characteristics. On the other hand, in a multi-dataset approach, merging the datasets may introduce some noise if the model does not have enough samples to weight the particular traits of the tested data.

4.3 Cross-Domain Settings: Testing on a Completely Unseen Dataset

The hypothesis behind the model generalization in machine learning, is its performance over the test split which stays unseen during the training process. However, this assumption has a couple of caveats based on the fact that we are drawing our test samples identically from the same distribution, and thus we are not biasing ourselves in any way [23]. Hence, and in order to answer the question: to which extent is our approach independent of the domain and data diversity, we adopt another examination of the model robustness over shifted or cross-domain settings. That is to say, we are testing on a completely new corpus and not only a subset of unseen samples from the same training corpus. Consequently, this approach is also known as *out-of-domain* testing. However, it has

Table 9. SDL vs. MDL argument unit classification using the stacked model.

		W-Precision		**W-Recall**		**W-F1 score**		**Accuracy**	
	Dataset	Mean	Std	Mean	Std	Mean	Std	Mean	Std
SDL	SE	**0.825**	±0.003	**0.827**	±0.003	**0.826**	±0.003	**0.827**	±0.003
	WD	**0.888**	±0.012	**0.868**	±0.01	**0.878**	±0.007	**0.868**	±0.01
	IBM	**0.987**	±0.002	**0.987**	±0.002	**0.987**	±0.002	**0.987**	±0.002
MDL	SE	0.736	±0.134	0.738	±0.125	0.737	±0.127	0.738	±0.125
	WD	0.802	±0.026	0.796	±0.015	0.799	±0.014	0.796	±0.015
	IBM	0.913	±0.006	0.895	±0.008	0.904	±0.009	0.895	±0.008

been referred to as cross-domain in different argument mining studies (e.g., [9]). Therefore, we apply our experiments in a hold-out manner. In other words, we keep out in each run one dataset for testing and we train on the remaining ones. We again assay in these experiments our stacked model with only structural features for argument unit classification and with structural, lexical and syntactical features for argument identification task (cf. Sect. 4.1). We report the weighted mean and standard deviation over 5 different seeds.

The results of cross-domain argument identification and cross-domain argument unit classification are presented in Table 10 and Table 11 respectively.

Table 10. Evaluation of the cross-domain argument identification task.

			W-Precision		**W-Recall**		**W-F1 score**		**Accuracy**	
Training	Testing	Model	Mean	Std	Mean	Std	Mean	Std	Mean	Std
SE	WD	stacked model	0.559	±0.006	0.455	±0.013	0.502	±0.013	0.455	±0.013
		DistilBERT	**0.661**	±0.003	**0.694**	±0.003	**0.677**	±0.002	**0.694**	±0.003
		[9]					0.524		0.524	
WD	SE	stacked model	0.749	±0.006	0.771	±0.012	0.760	±0.006	0.771	±0.012
		DistilBERT	**0.759**	±0.006	**0.798**	±0.005	**0.778**	±0.004	**0.798**	±0.005
		[9]					0.128		0.181	

In terms of argument identification task, and based on the empirical evaluation presented in Table 10, we observe a satisfactory performance of our stacking model (W-F1 score= 0.76) when training on WD and testing on SE. However, the opposite scenario drastically reduces the performance where (W-F1 score= 0.502). While those are both better than the outcomes of [9] who used a binary statistical classifier with a similar set of our SVM features, DistilBERT is still able to outperform our stacking model in this scenario.

Table 11. Evaluation of the cross-domain argument unit classification task.

			W-Precision		**W-Recall**		**W-F1 score**		**Accuracy**	
Training	Testing	Model	Mean	Std	Mean	Std	Mean	Std	Mean	Std
SE, WD	IBM	stacked model	**0.766**	±0.015	**0.610**	±0.052	**0.679**	±0.081	**0.61**	±0.052
		DistilBERT	0.704	±0.028	0.550	±0.013	0.618	±0.024	0.55	±0.013
SE, IBM	WD	stacked model	0.735	±0.08	0.546	±0.281	0.627	±0.303	0.546	±0.281
		DistilBERT	**0.773**	±0.008	**0.805**	±0.009	**0.789**	±0.004	**0.805**	±0.009
WD,IBM	SE	stacked model	**0.677**	±0.013	**0.675**	±0.016	**0.676**	±0.044	**0.675**	±0.016
		DistilBERT	0.356	±0.128	0.586	±0.128	0.443	±0.141	0.586	±0.128

With regards to the argument unit classification (Table 11), we observe that training the stacked model on SE plus IBM and testing on WD yields worse results than training on other datasets (W-F1 score=0.627). However, it is still outperforming DistilBERT when testing on IBM and SE. In fact, the performance of DistilBERT degraded for this task, especially when testing on SE, and it achieves its best performance when testing on WD. That means, for premise/claim classification, we still need the features of SVM that allow our stacked model to overcome transfer learning once the tested corpus implies a formal structure that could be better learned using traditional machine learning. This also interprets the worst case of stacked model (trained on SE, IBM and tested on WD), since WD does not imply such learned features (e.g., sentence position) and by contrary SVM pulls back the stacked model performance in this testing scenario.

To sum up this section, our results suggest that transferring knowledge across different datasets is more applicable for argument identification task. Comparing to [11], DistilBERT is still reaching a higher accuracy when fine-tuned on the same dataset. This means that transfer learning is very efficient for in-domain-generalization, and less efficient for cross or out-of-domain generalization. However, this is even more challenging for argument unit classification where our stacking approach shows a better generalizing capability, in most cases, with the power of learning genre-independent presentations of argument units. We further apply cross-topic testing in Sect. 4.4.

4.4 Cross-Topic Settings: Testing on Unseen Topics from the Same Dataset(s)

In this section, we further assess the stacked model performance and compare it with DistilBERT, over unseen data, with a finer-grained level of cross-settings referred to as cross-topic. In this experiment, we aim to study whether the model performance over unseen topics will be improved by considering more training topics, or by considering more samples for each training topic. In other words, the analysis will reveal whether or not the diversity sampling (a wide range of topics) improves cross-topic performance.

Data. To perform these experiments, we derive a group of new datasets out of the SE, WD, and IBM datasets according to each particular classification task. The number of sentences per topic (#S/T) varies across the three datasets. However, we still need to unify the size of data for all tested combinations, as well as unifying the #S/T in every combination. By that, we analyse only the effect of diversity sampling (#T) on the model generalization to unseen topics. Hence, we fix the dataset size to 1200 sentences in each experiment. This is the maximum possible size of data that gives different combinations of #T and #S/T with respect to our corpora statistics. Apparently, this implies that a higher number of topics leads to a lower number of sentences per topic.

Experimental Set-Up. To generalize our results, We run the cross-topic over 5 runs (5 seeds) and internally over a 5-fold cross-validation setup. We report the average mean and standard deviation of the weighted precision, recall, F1 score, and accuracy on the testing set. Our 5-fold cross-validation is in terms of topics. In other words, the training set covers 80% of the topics and the remaining unseen topics are in the testing set.

Evaluation. In the following, we present the obtained results in Table 12 and Table 13 for argument identification and argument unit classification respectively. For argument identification, the evaluation results prove that the stacking model performance is consistent over the different sets of topics: W-F1 score averages between 0.810 to 0.893, and the accuracy ranges from 0.813 to 0.895. Similarly, in the unit classification task, the W-F1 averages between 0.801 to 0.858, and accuracy ranges from 0.80 to 0.855.

These findings suggest that the ensemble learning stacking approach is outperforming DistilBERT in all the cases with W-F1 score approximately +10% for argument identification and up to +5% for argument unit classification. Moreover, the former reported a lower variance in the standard deviation for almost all tested cases. This is in line with the findings of [10] who found that 100 instances of BERT is remarkably consistent in their in-distribution generalization accuracy, while they varied dramatically in their out-of-distribution generalization performance. Therefore, since BERT-like model (DistilBERT in our case) is less stable to completely unseen data, the stacked approach gets a valuable impact on the model robustness in such out-of-distribution or cross-domain scenarios. Moreover, according to Zhang et al. [35], BERT only exploit plain context-sensitive features such as character or word embeddings. It rarely consider incorporating structured semantic information which can provide rich semantics for language representation.

In terms of the impact of #T and #S/T, the weighted F1 score has been improved by increasing the #T in the training set for the argument identification task. However, the opposite behavior is observed concerning the argument unit classification task: i.e., increasing the #T decreased the weighted F1 score. We explain this contrast by the influence of the vocabulary employed in each task. In fact, the structure of arguments may differ according to the discussed topic.

Table 12. Model assessment in cross-topic experiments for argument identification task. #S/T: number of Sentences/Topic, #T: number of Topics

			W-Precision		**W-Recall**		**W-F1 score**		**Accuracy**	
#S/T	#T	Model	Mean	Std	Mean	Std	Mean	Std	Mean	Std
4	300	stacked model	**0.892**	±0.019	**0.895**	±0.019	**0.893**	±0.021	**0.895**	±0.019
		DistilBERT	0.765	±0.083	0.825	±0.03	0.794	±0.052	0.825	±0.03
6	200	stacked model	**0.855**	±0.009	**0.862**	±0.008	**0.858**	±0.009	**0.862**	±0.008
		DistilBERT	0.703	±0.089	0.791	±0.021	0.744	±0.036	0.791	±0.021
24	50	stacked model	**0.807**	±0.029	**0.813**	±0.026	**0.81**	±0.032	**0.813**	±0.026
		DistilBERT	0.626	±0.09	0.775	±0.03	0.693	±0.041	0.775	±0.03

For instance, we can find more statistical arguments in finance and more logical well-structured arguments in law. Therefore, ensuring distinct and diverse samples (varying topics during the training process) is important to generalize the learned patterns of argumentative text. However, for the argument unit classification, distinguishing between premise and claim is more related to the grammatical structure of sentences which does not require topic-specific vocabulary. For instance, we can use claim keywords (consequently, in fact, implies) or premise keywords (such as because, moreover, since) to distinguish between the argument components.

Table 13. Model assessment in cross-topic experiments for argument unit classification task. #S/T: number of Sentences/Topic, #T: number of Topics

			W-Precision		**W-Recall**		**W-F1 score**		**Accuracy**	
#S/T	#T	Model	Mean	Std	Mean	Std	Mean	Std	Mean	Std
3	400	stacked model	**0.802**	±0.013	**0.8**	±0.014	**0.801**	±0.013	**0.8**	±0.014
		DistilBERT	0.774	±0.02	0.767	±0.023	0.77	±0.022	0.767	±0.023
4	300	stacked model	**0.822**	±0.03	**0.82**	±0.032	**0.821**	±0.034	**0.82**	±0.032
		DistilBERT	0.764	±0.032	0.766	±0.031	0.765	±0.031	0.766	±0.031
6	200	stacked model	**0.825**	±0.019	**0.825**	±0.019	**0.825**	±0.02	**0.825**	±0.019
		DistilBERT	0.789	±0.02	0.786	±0.019	0.787	±0.019	0.786	±0.019
24	50	stacked model	**0.861**	±0.054	**0.855**	±0.055	**0.858**	±0.056	**0.855**	±0.055
		DistilBERT	0.847	±0.074	0.835	±0.079	0.841	±0.076	0.835	±0.079

5 Conclusion

We address in this paper two main problems of argument mining: argument identification and argument unit classification. Our study is on the sentence-level with a stacked ensemble learning approach. We aim to detect the essence of argumentative text and to assess the robustness of our model in more realistic scenarios than testing on a subset of the data known as the test split.

Furthermore, while generalization has always been an important research topic in machine learning research, the robustness and generalization of argument mining models are yet not well explored. This is a very urgent task to elevate the research in this field given the two-fold challenge it has: the lack of labeled data, and the domain dependency performance of the existing models. We believe that a formal protocol of testing the model generalization and robustness is an instant need in argumentation domain since every paper tackles it from only one angle. Most of the works suggest cross-domain models with the mean of integrating more datasets in the training process.

Therefore, in this paper, we defined sets of experiments that infer an empirical evidence on the model performance in real world applications. Based on our comparison of single-dataset learning (SDL) and multi-dataset learning (MDL), we propose that SDL is always recommended when we are confident that future dataset will be similar to the training one. Furthermore, our findings suggest that knowledge transfer is more applicable for argument identification than argument unit classification in cross-domain (out-of-distribution) setup. In terms of the latter task, the stacked model outperformed DistilBERT when tested on IBM and SE corpora. This indicates that recognizing premise and claim texts is more related to the structure of the sentence. A similar conclusion is reached in our cross-topic experiments on this particular task, where we found that the more #S/T (number of sentences per topic) we have for training, the better the stacked model generalizes to unseen topics. However, the sampling diversity (increasing the topic count #T) was essential for the argument identification task such that topic-specific vocabulary plays a crucial role.

Since the structure of the sentence made a difference in many of our experiments, we plan to test if providing a transfer learning approach (e.g., DistilBERT) with such features, would outperform the ensemble learning approach based on this enriched knowledge. This research direction is towards the understanding of how transformers indeed work, and how we can develop them [36]. In our future work, we also plan to run joint model experiments where argument identification and argument component classification are in one sequential pipeline. We also plan to investigate more on the segmentation model that predicts the boundaries of the argument and on optimizing the combination of the base models (SVM and DistilBERT).

Acknowledgements

The project on which this report is based was partly funded by the German Federal Ministry of Education and Research (BMBF) under the funding code 01—S20049. The author is responsible for the content of this publication.

References

1. Baker, A.: Simplicity, the Stanford Encyclopedia of Philosophy. Metaphysics Research Lab (2016)
2. Lawrence, J., Reed, C.: Argument mining: a survey. Comput. Linguist. **45**(4), 765–818 (2020)
3. 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 (2009)
4. Toulmin, S.E.: The Uses of Argument. Cambridge University Press, Cambridge (2003)
5. Stab, C., Gurevych, I.: Annotating argument components and relations in persuasive essays. In Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics. Technical papers, pp. 1501–1510 (2014)
6. Song, Y., Heilman, M., Klebanov, B.B., Deane, P.: Applying argumentation schemes for essay scoring. In: Proceedings of the First Workshop on Argumentation Mining, pp. 69–78 (2014)
7. Samadi, M., Talukdar, P., Veloso, M., Blum, M.: Claimeval: integrated and flexible framework for claim evaluation using credibility of sources. In: Thirtieth AAAI Conference on Artificial Intelligence (2016)
8. Alhamzeh, A., Bouhaouel, M., Egyed-Zsigmond, E., Mitrovic, J.: Distilbert-based argumentation retrieval for answering comparative questions. In: Working Notes of CLEF (2021)
9. Al-Khatib, K., Wachsmuth, H., Hagen, M., Köhler, J., Stein, B.: Cross-domain mining of argumentative text through distant supervision. In: Proceedings of NAACL-HLT, pp. 1395–1404 (2016)
10. McCoy, R.T., Min, J., Linzen, T.: BERTs of a feather do not generalize together: Large variability in generalization across models with similar test set performance. In: Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP, pp. 217–227. Association for Computational Linguistics (2020)
11. Alhamzeh, A., Bouhaouel, M., Egyed-Zsigmond, E., Mitrović, J., Brunie, L., Kosch, H.: A stacking approach for cross-domain argument identification. In: Strauss, C., Kotsis, G., Tjoa, A.M., Khalil, I. (eds.) DEXA 2021. LNCS, vol. 12923, pp. 361–373. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-86472-9_33
12. Liga, D., Palmirani, M.: Transfer learning with sentence embeddings for argumentative evidence classification (2020)
13. Wambsganss, T., Molyndris, N., Söllner, M.: Unlocking transfer learning in argumentation mining: a domain-independent modelling approach. In: 15th International Conference on Wirtschaftsinformatik (2020)

14. Zhang, W.E., Sheng, Q.Z., Alhazmi, A., Li, C.: Adversarial attacks on deep-learning models in natural language processing: a survey. ACM Trans. Intell. Syst. Technol. (TIST) **11**(3), 1–41 (2020)
15. Tzeng, E., Hoffman, J., Saenko, K., Darrell, T.: Adversarial discriminative domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7167–7176 (2017)
16. Schiller, B., Daxenberger, J., Gurevych, I.: Stance detection benchmark: how robust is your stance detection? KI - Künstl. Intell. **35**(3), 329–341 (2021). https://doi.org/10.1007/s13218-021-00714-w
17. Ajjour, Y., Chen, W.F., Kiesel, J., Wachsmuth, H., Stein, B.: Unit segmentation of argumentative texts. In: Proceedings of the 4th Workshop on Argument Mining, pp. 118–128 (2017)
18. Al Khatib, K., Wachsmuth, H., Kiesel, J., Hagen, M., Stein, B.: A news editorial corpus for mining argumentation strategies. In Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pp. 3433–3443 (2016)
19. Habernal, I., Gurevych, I.: Argumentation mining in user-generated web discourse. Comput. Linguist. **43**(1), 125–179 (2017)
20. Bouslama, R., Ayachi, R., Amor, N.B.: Using convolutional neural network in cross-domain argumentation mining framework. In: Ben Amor, N., Quost, B., Theobald, M. (eds.) SUM 2019. LNCS (LNAI), vol. 11940, pp. 355–367. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-35514-2_26
21. Elangovan, A., He, J., Verspoor, K.: Memorization vs. generalization: quantifying data leakage in NLP performance evaluation. In: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, pp. 1325–1335 (2021)
22. Huan, X., Mannor, S.: Robustness and generalization. Mach. Learn. **86**(3), 391–423 (2012). https://doi.org/10.1007/s10994-011-5268-1
23. Wang, J.: Generalizing to unseen domains: a survey on domain generalization. IEEE Trans. Knowl. Data Eng. (2022)
24. Stab, C., Gurevych, I.: Parsing argumentation structures in persuasive essays. Comput. Linguist. **43**(3), 619–659 (2017)
25. Stab, C.: Argumentative Writing Support by Means of Natural Language Processing, p. 208 (2017)
26. Aharoni, E.: 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, pp. 64–68 (2014)
27. Sagi, O., Rokach, L.: Ensemble learning: a survey. Wiley Interdis. Rev.: Data Min. Knowl. Discovery **8**(4), e1249 (2018)
28. 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 (2007)
29. Stab, C., Gurevych, I.: Identifying argumentative discourse structures in persuasive essays. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 46–56 (2014)
30. Sanh, V., Debut, L., Chaumond, J., Wolf, T.: Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. arXiv preprint. arXiv:1910.01108 (2019)
31. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: pre-training of deep bidirectional transformers for language understanding. arXiv preprint. arXiv:1810.04805 (2018)

32. Ryu, M., Lee, K.: Knowledge distillation for bert unsupervised domain adaptation. arXiv preprint. arXiv:2010.11478 (2020)
33. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint. arXiv:1711.05101 (2017)
34. De Winter, J.C.F.: Using the student's t-test with extremely small sample sizes. Pract. Assess. Res. Eval. **18**(1), 10 (2013)
35. Zhang, Z., et al.: Semantics-aware BERT for language understanding. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, no. 05, pp. 9628–9635 (2020)
36. Rogers, A., Kovaleva, O., Rumshisky, A.: A primer in bertology: what we know about how bert works. Trans. Assoc. Comput. Linguist. **8**, 842–866 (2020)