



Methods for Automatic Argumentation Structure Prediction

Ilya Dimov^{1(✉)} and Boris Dobrov^{1,2(✉)}

¹ Lomonosov Moscow State University, Moscow, Russia
iliyadimov@icloud.com, dobrov.bv@mail.ru

² Research Computing Center of M.V. Lomonosov Moscow State University,
Moscow, Russia

Abstract. Argumentation mining is a natural language understanding task consisting of several subtasks: relevance detection, stance classification, argument quality assessment and fact checking. In this work we propose several architectures for the analysis of argumentative texts based on BERT. We also show, that models, which jointly learn argumentation mining subtasks outperform pipelines of models trained on a single tasks. Additionally we explore transfer learning approach based on pretraining for the natural language inference task, which achieves highest score on tasks of argumentation mining among the models trained on english corpora.

Keywords: Argumentation mining · Relevance detection · Stance classification

1 Introduction

Argumentation is the process of forming reasons and of drawing conclusion and applying them to a case in discussion. The task of argumentation mining is aimed at the extraction of the argument components and linking them in a structure.

There are various ways to construct the argument structure, varying in the number of components and types of connections between them. In this work, following the notion introduced in the works of IBM Project Debater¹ [8, 11, 19], we use three entities:

- *Topic* - a short, concise and controversial statement, which serves to frame the discussion.
- *Claim* - a statement, that directly supports or contests the topic.
- *Evidence* - a statement, that directly supports or contests the topic, which is not a mere belief or a claim.

We additionally define a *focus* as the subject mentioned in the *topic*. There are two tasks, that help to restore the structure of the argument:

¹ https://www.research.ibm.com/haifa/dept/vst/debating_data.shtml.

1. Relevance detection - given a topic and a sentence, find out whether the sentence can be used as claim or evidence.
2. Stance classification - given a topic and a valid claim or evidence, classify the stance as a support or rebuttal (PRO/CON).

These tasks are hard to solve, as the texts often do not provide full context of the underlying discussion. For example, let's take the following pair of a topic and a claim:

Topic: We should further exploit nuclear power
 Claim: Similarly, after the Chernobyl incident, ... they
 believed a similar meltdown was likely to happen in the U.S.

There is no lexical intersection between the topic and the claim. To correctly link those two entities the system must understand, that Chernobyl is connected to nuclear energy. One way to solve this problem is to supply the required facts to the model as an additional input. Another approach is to use a large pretrained model. BERT [7] has shown great success in various tasks of natural language understanding such as question answering and commonsense reasoning [6].

In this work we explore several architectures based on BERT and a transfer learning approach and try to analyze the model outputs and understand, how to further improve its results. The code for our experiments is publically available².

2 Related Work

The task of argument mining was explored by a number of various approaches. For example systems MARGOT [12] and Targer [5] the task is formulated as a sequence tagging problem. MARGOT's pipeline first classifies sentences as containing claim or evidence with respect to a topic, before extracting a continuous span of text, which serves as claim or evidence. Similiar task is solved in [2], where a collection of phrases classified as argumentative or non-argumentative. Sequence-tagging approach is harder than sentence-level classification, because it requires a more complex dataset, which is difficult to construct.

The paper [15] proposes another corpora, where sentences are classified with respect to the focus as supporting, opposing and not relevant arguments.

In the work [1] the structure of the argument is different: evidence is connected to claims and claims are connected to topics, instead of the claim and the evidence being both connected to the topic. The proposed dataset has additional markup of evidence to 3 types, based on its origin: Anecdotal, Expert and Study.

In [11, 14] authors describe a method of automatic claim mining. They gather a weak labeled claim corpora and show, that it can be used for model improvement alongside human-annotated dataset.

Another subtask of argumentation mining is argumentation quality assessment [9, 10, 18]. The task is, given a claims or evidence relevant to the topic, to arrange them in the way that the most strong arguments are first. The are two

² <https://github.com/hawkeoni/Argument-structure-prediction>.

main approaches: first focuses on setting a value from 0 to 1 for an argumentation component, while the second approach consists of selecting the strongest claim or evidence phrases with respect to the topic out of several options.

As stated in the definition, evidence must not be a mere belief or a baseless statement. For example study-type evidence may often refer to some research, but such research may not exist, thus fooling the system. To battle this the task of fact checking is introduced. In CLEF-2020 CheckThat! [3] lab several subtasks are proposed, based on the data from Twitter:

- Check Worthiness - given a topic and a stream of potentially-related tweets, rank the tweets according to their check-worthiness for the topic. A check-worthy tweet is a tweet that includes a claim that is of interest to a large audience (specially journalists), might have a harmful effect, etc.
- Claim Retrieval - Given a check-worthy claim and a dataset of verified claims, rank the verified claims, so that those that verify the input claim (or a sub-claim in it) are ranked on top.
- Evidence Retrieval - Given a check-worthy claim on a specific topic and a set of text snippets extracted from potentially-relevant webpages, return a ranked list of evidence snippets for the claim. Evidence snippets are those snippets that are useful in verifying the given claim.

Similar problem is stated in FEVER dataset [17]: given a claim the task is to find relevant Wikipedia sentences, which support it.

Another similar task is natural language inference or textual entailment. Given a text and a hypothesis the goal is to determine, whether the hypothesis is true. While the task is not directly connected to argumentation mining it is highly similar to it. Hypothesis can be entailed by text, contradict it or not be relevant to the text, just as claims or evidence can be relevant and not relevant to the topic.

3 Model Architectures

We formulate tasks of stance classification and relevance detection as binary classification problems and solve them with three architectures, based on the english version of BERT-base model from [7]. All architectures take a pair sentences as input: a topic and a claim or a topic and evidence, and then predict the probability of those components being related, or the probability of the support relation depending on the task.

BERT is a transformer encoder [20] built on the self-attention mechanism. We use $BERT_{base}$ model weights, which consist of 12 transformer blocks. Each block calculates multihead self attention over the inputs (either token embeddings or previous layer outputs):

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \quad (1)$$

where Q , K , V are linear projections of the inputs, d_k is the dimension of attention head. The self-attention layer is followed by a layer normalization of sum of the input and the resulting self attention activations, which is called residual connection. Normalized activations are fed to a feedforward layer, which adds nonlinearity to the model:

$$FeedForward(x) = \max(0, xW_1 + b_1)W_2 + b_2 \quad (2)$$

The feedforward layer is followed by another normalization over a residual connection.

The original BERT model is trained on a masked language model task, where it has to predict masked tokens in the sentence by leveraging the context from all tokens, as opposed to RNN-based models, which employ either the left or the right context or rely on bidirectional approach where two RNNs are used. Additionally authors use the task of next sentence prediction, where the model also predicts, if two sentences follow each other.

The task of masked language modeling provides a good starting point for model finetuning of various downstream tasks such as named entity recognition, natural language inference, question answering and many other, as stated in the original work [7].

We formulate the subtasks of argumentation mining as classification and solve them the following way: we encode pairs of sentences (topic and claim, topic and evidence) via BERT and use this rich contextual representation to classify the relation between texts.

Our first architecture *BERT_{basic}* follows a standart approach for classification originally used for next sentence prediction. We concatenate the text in the following way:

$$[CLS] \text{ text}_1 [SEP] \text{ text}_2 [SEP] \quad (3)$$

where $[CLS]$ is a special token representing the start of the text and $[SEP]$ represents the end of the text. The final vector representation of the $[CLS]$ token is used in a linear layer followed by a softmax over classes for classification. This is a basic approach for text classification with BERT, which makes the model learn a suitable representation of the text for the downstream task.

However the computational complexity of self-attention layer is $O(N^2)$, where N is the length of the input. In the general task of argumentation mining it is required given many topics and texts to extract the relevant pieces of information, which means that we have to calculate the outputs of the model for each possible pair of argumentation components. Thus we propose a second approach, we call *BERT_{vect}*. We calculate the following vector representations ($[:,]$ represents vector concatenation):

$$x1 = BERT([CLS] \text{ text}_1 [SEP]) \quad (4)$$

$$x2 = BERT([CLS] \text{ text}_2 [SEP]) \quad (5)$$

$$class = softmax([x1; x2]W + b) \quad (6)$$

This approach allows precomputation of vector representations of text, which can be later used in a simple linear model. While this approach is faster, it lacks the cross-sentence attention, which was previously available in $BERT_{basic}$ architecture.

Our final model is $BERT_{se}$, based on the architecture proposed in [16]. It is mainly similar to $BERT_{basic}$ model, but it adds a special self-explaining layer at the end. After the argumentation components are passed through BERT and enriched with contextual information via attention mechanism, the self-explaining layer extracts span representations $h(i, j)$ in the following way:

$$h(i, j) = \tanh[W(h_i, h_j, h_i - h_j, h_i \otimes h_j)] \quad (7)$$

$$o(i, j) = \hat{h}^T h(i, j) \quad (8)$$

$$\alpha(i, j) = \frac{\exp o(i, j)}{\sum \exp(o(i, j))} \quad (9)$$

$$\tilde{h} = \sum \alpha(i, j) h(i, j) \quad (10)$$

$W, \hat{h}$ represent learnable parameters, h_i, h_j represent the output on the last layer of BERT corresponding to the i -th and the j -th token. The final prediction is made on the $\tilde{h}$, which gathers the most important span information, instead of the [CLS] token used in $BERT_{basic}$.

We also follow the modification of loss proposed in the original work:

$$Loss = \log p(y|x) + \lambda \sum_{i,j} \alpha^2(i, j) \quad (11)$$

The increase of the λ factor makes the span α distribution sharper, which in turn makes model focus more on a single span.

Additionally we test pretaining on SNLI (Stanford Natural Language Inference corpus) [4] corpora as a strategy for improving quality on argumentation mining subtasks. The corpora consists of pairs of text and hypothesis and the task is to classify whether the hypothesis contradicts, entails or bears no relevance to the text.

While SNLI mostly contains very simple premises which do not require external knowledge, we believe that its properties may improve model behaviour on both the stance classification and relevance detection as it will help model to better extract features required for general purpose natural language understanding. The SNLI corpora is very rich in paraphrases which may help to better process lexically distant pairs of argumentation components.

4 Datasets

In this work we explore the datasets ArgsEN and EviEN introduced in [19]. The ArgsEN corpora introduces two tasks: stance classification and argument quality assessment. In this work we only use the task of stance classification, which consists of 30497 pairs of topic and claim (Fig. 1).

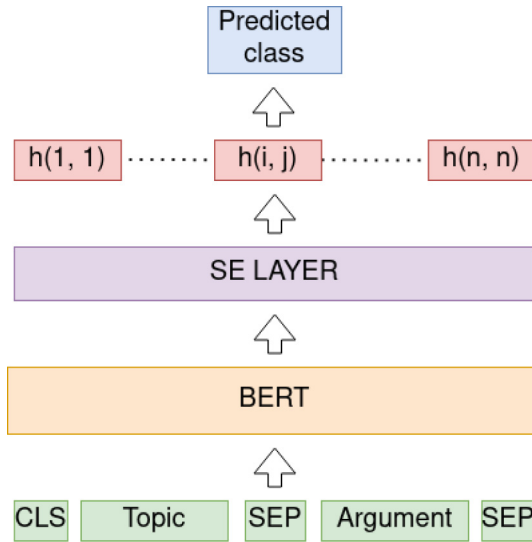


Fig. 1. $BERT_{SE}$ model architecture.

Topic: We should end the use of economic sanctions
 Claim: Economic sanctions causes safer trading of goods
 Stance: PRO

Topic: We should ban algorithmic trading
 Claim: algorithmic trading can do a job that humans
 are not able to do
 Stance: CON

EviEN has two tasks: stance classification and evidence detection (relevance detection). The first task is similar to ArgsEN - given a pair of topic and evidence determine, whether evidence supports or contests the topic. The goal of relevance detection is to determine, whether the evidence is relevant to the topic or not. Combined with the task of stance classification it constitutes for a complete argumentation structure prediction pipeline. The EviEN corpora consists of 35211 samples, all of which are marked up for relevance and 10381 marked for stance.

Topic: We should ban the sale of violent video games to minors
 Evidence: The storylines of games in this subgenre typically
 have strong themes of crime and violence.
 Relevance: Relevant
 Stance: PRO

Topic: We should legalize doping in sport
 Evidence: Contador signed a commitment in which he stated:

"I am not involved in the Puerto affair nor in any other doping case"
 Relevance: Not relevant
 Stance: Not applicable

The train-test split in ArgsEN and EviEN is proposed by the authors in such a way, that training data has no topic overlap with the test.

We also explore the transferability of the natural language inference on the tasks of relevance detection and stance classification. As stated before, these problems are similar in formal definition: both tasks are formulated as classification of pairs of statements have similar classes. We use the Stanford NLI [4] to test, whether pretraining the task of textual entailment can improve the quality of argumentation mining models.

5 Experiments

We are using the english weights of $BERT_{base}$ model in all our experiments. Our training hyperparameters also remain the same across all experiments: the models were finetuned for 5 epochs with the learning rate of $2e-5$ on effective batch size of 32. The computations were made on a single Nvidia Tesla V100 GPU using gradient accumulation for $BERT_{SE}$ architecture with the parameter λ of 1.

Table 1. Model metrics on test splits of different subtasks. The values in the “ArgsEN stance” column are the macro-F1 and the accuracy metrics. The values in the “EviEN combined” columns are accuracy metrics for a combined model and a pipeline of models.

Model	ArgsEN stance	EviEN relevance	EviEN stance	EviEN combined
$IBMBERT_{en}$	89.3	—	—	—
$IBMBERT_{17L}$	91.5	—	—	—
$BERT_{basic}$	89.9 (90.)	69.1	86.8	64.7 (52.5)
$BERT_{vect}$	74.5 (74.5)	64.7	72.1	52.3 (46.3)
$BERT_{SE}$	90.0 (90.1)	71.6	87.6	67.4 (56.1)
$SNLI_{basic+ZS}$	56.2 (59.0)	45.6	50.4	30.8
$SNLI_{SE+ZS}$	62.3 (62.5)	46.5	57.1	32.3
$SNLI_{basic+FT}$	89.0 (89.0)	71.9	88.4	63.3
$SNLI_{SE+FT}$	90.6 (90.6)	69.1	88.8	64.3

As stated previously the task of stance classification is, given a topic and a relevant text, to classify whether the text supports the topic or contradicts it. We use the ArgsEN dataset, introduced in [19]. We also adapt the metric in the original work, which is macro-F1 averaged by the two classes of PRO/CON. Additionally we calculate accuracy on the whole test split of the dataset.

The authors of [19] provide several models based on the weights of BERT-multilingual model. Those models have *IBM* prefix in the Table 1. The model *IBMBERT*_{17L} was trained on a multilingual stance classification corpora, which greatly improved its performance on the english version of the corpora making it currently the best performing model. However, we train our models only on the english data, so our models should be compared to the *IBMBERT*_{en} model, which was also trained on monolingual corpora.

Models *BERT*_{basic}, *BERT*_{SE} show performance improvement compared to the models in original work, which may be attributed to the english version of the BERT weights. *BERT*_{SE} model does not significantly increase model quality over *BERT*_{basic}, while *BERT*_{vect} displays a relatively low metric.

We train our best performing architectures *BERT*_{basic} and *BERT*_{SE} on the SNLI corpora [4] and explore the capabilities of zero-shot approach. The models using zero-shot technique have suffix *ZS* in their name in the Table 1. For the tasks of stance classification we excluded the “neutral” or “no relation” class from the models prediction. This approach yields negative results as the metrics do not significantly exceed random guess. We additionally explore finetuning after the SNLI pretraining, which greatly improves upon models trained solely on the stance classification task. Such models have *FT* suffix in the Table 1. The finetuning procedure uses the same hyperparameters as the training, but introduces a newly initialized final linear layer.

We also conduct the same experiments on the stance classification task on the EviEN corpora. The experiments yield similiar results - relative model performance remains identical across the two datasets.

We conduct the same experiments on the tasks of evidence detection on the EviEN corpora as on the tasks of stance classification. The authors of the datasets [19] provide no baselines for the task.

Our results align with the experiments on the stance classification, although the quality of relevance detection is generally much lower. During SNLI pretraining and finetuning we combine the classes of “entailment” and “contradiction”, because both those classes imply a relation between the argumentation components.

Finally we explore, how our models perform on a full argumentation mining pipeline. We call this “EviEN combined” task - the goal is, given a topic and a possible evidence text, to classify if it is relevant to the text and further classify its stance if it is indeed relevant.

Our first approach consists of training a model on the combined labels of the EviEN corpora, so the models directly classifies the textual pairs relations as “not relevant”, “entailment” and “contradiction”. Our second approach consists of pipelining previously trained models: firstly we predict the relevance and only for relevant pairs we predict the stance. As the table shows, combined pipeline improves overall performance on the argumentation mining task.

Despite this setup being the closest to SNLI, as no classes were changed or ignored, because there is a direct mapping between the classes of textual

entailment and the 3 classes used in the full argumentation mining pipeline, this approach did not improve the quality of the combined task.

Overall, the experiment results are:

- The $BERT_{basic}$ and $BERT_{SE}$ architectures provide strong baselines for the subtasks of argumentation mining with the latter generally outperforming other models by a margin of several points.
- $BERT_{vect}$ model quality indicates, that the mechanism of cross-sentence attention is highly beneficial to the metrics in the tasks of argumentation mining, when using BERT weights. However it may still be possible to achieve higher computational speed without significant quality loss by using specially trained sentence embeddings, such as [13].
- Zero shot approach based on SNLI pretraining yields no result, as the quality does not noticeably exceed random guess, but SNLI pretraining with further finetuning results in a noticeable improvement over models trained exclusively on the tasks of stance classification and relevance detection.
- The combined approach, trained simultaneously on the tasks of stance classification and evidence detection strongly outperforms pipeline of two models trained on singular tasks.

6 Conclusion

In this work we propose several strong baselines for the subtasks of argumentation mining: stance classification and relevance detection. We find out that cross-sentence attention mechanism is required for a successful model approach using the BERT architecture. Our strongest performing model is $BERT_{SE}$, which was adapted from the natural language inference task and improves upon $BERT_{basic}$ architecture by amplifying cross-sentence attention mechanism with an additional interpretation layer.

We also explore the possibility of combining the subtasks of argumentation mining into classification problem with 3 classes. Models trained using this approach outperform a pipeline of two models for two subtasks on the EviEN dataset.

We also explore the approach of transferring knowledge from SNLI corpora to the ArgsEN and EviEN datasets, which improves model quality and achieves the highest metric on the task of stance classification among the models trained on english data. Additionally we try this technique in a zero shot scenario, but it yields poor results.

We believe that future work might address the problems of exploration of the model’s inner knowledge as well as the problem of ways to directly supply model with external information about the world.

Acknowledgements. The study is supported by Russian Science Foundation, project 21-71-30003.

References

1. Aharoni, E., et al.: 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)
2. Al-Khatib, K., Wachsmuth, H., Hagen, M., Stein, B., Köhler, J.: Webis-debate-16. Zenodo, January 2016. <https://doi.org/10.5281/zenodo.3251804>
3. Barrón-Cedeño, A., et al.: Overview of CheckThat! 2020: automatic identification and verification of claims in social media. In: Arampatzis, A., et al. (eds.) CLEF 2020. LNCS, vol. 12260, pp. 215–236. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58219-7_17
4. Bowman, S.R., Angeli, G., Potts, C., Manning, C.D.: A large annotated corpus for learning natural language inference. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP). Association for Computational Linguistics (2015)
5. Chernodub, A., et al.: TARGER: neural argument mining at your fingertips. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pp. 195–200 (2019)
6. Cui, L., Cheng, S., Wu, Y., Zhang, Y.: Does BERT solve commonsense task via commonsense knowledge? arXiv preprint [arXiv:2008.03945](https://arxiv.org/abs/2008.03945) (2020)
7. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. arXiv preprint [arXiv:1810.04805](https://arxiv.org/abs/1810.04805) (2018)
8. Ein-Dor, L., et al.: Corpus wide argument mining-a working solution. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 7683–7691 (2020)
9. Gleize, M., et al.: Are you convinced? Choosing the more convincing evidence with a Siamese network. arXiv preprint [arXiv:1907.08971](https://arxiv.org/abs/1907.08971) (2019)
10. Gretz, S., et al.: A large-scale dataset for argument quality ranking: construction and analysis. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 7805–7813 (2020)
11. Levy, R., Bogin, B., Gretz, S., Aharonov, R., Slonim, N.: Towards an argumentative content search engine using weak supervision. In: Proceedings of the 27th International Conference on Computational Linguistics, pp. 2066–2081 (2018)
12. Lippi, M., Torroni, P.: MARGOT: a web server for argumentation mining. Expert Syst. Appl. **65**, 292–303 (2016)
13. Reimers, N., Gurevych, I.: Sentence-BERT: sentence embeddings using Siamese BERT-networks. arXiv preprint [arXiv:1908.10084](https://arxiv.org/abs/1908.10084) (2019)
14. Shnarch, E., et al.: Will it blend? Blending weak and strong labeled data in a neural network for argumentation mining. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 599–605 (2018)
15. Stab, C., Miller, T., Gurevych, I.: Cross-topic argument mining from heterogeneous sources using attention-based neural networks. arXiv preprint [arXiv:1802.05758](https://arxiv.org/abs/1802.05758) (2018)
16. Sun, Z., et al.: Self-explaining structures improve NLP models. arXiv preprint [arXiv:2012.01786](https://arxiv.org/abs/2012.01786) (2020)
17. Thorne, J., Vlachos, A., Christodoulopoulos, C., Mittal, A.: FEVER: a large-scale dataset for fact extraction and verification. arXiv preprint [arXiv:1803.05355](https://arxiv.org/abs/1803.05355) (2018)

18. Toledo, A., et al.: Automatic argument quality assessment-new datasets and methods. arXiv preprint [arXiv:1909.01007](https://arxiv.org/abs/1909.01007) (2019)
19. Toledo-Ronen, O., Orbach, M., Bilu, Y., Spector, A., Slonim, N.: Multilingual argument mining: datasets and analysis. arXiv preprint [arXiv:2010.06432](https://arxiv.org/abs/2010.06432) (2020)
20. Vaswani, A., et al.: Attention is all you need. In: Advances in Neural Information Processing Systems, pp. 5998–6008 (2017)