Unsupervised Argument Similarity via Sentence Compression

Siyi Liu
University of Pennsylvania
siyiliu@seas.upenn.edu

Abstract

Argument similarity is the task of determining if two arguments are similar. Previous methods based on contextualized embeddings fall short of identifying salient information in the arguments. We propose a re-weighting scheme of argument representations via sentence compression, and show that it outperforms previous state-of-the-art on a popular argument similarity benchmark. We further discuss the benefits and obstacles of using sentence compression model for argumentation tasks, and identify the limitations of contextual representations of arguments.

1 Introduction

When people converse about social or political topics, they often convey their opinions towards the topics using arguments. In order to learn the differences in attitudes towards a same topic among different authors, it is crucial to 1) identify their arguments made and 2) compare the similarities of their arguments. Previous works (Stab and Gurevych, 2014; Ein-Dor et al., 2019; Shnarch et al., 2018) have intensively studied the first step of the problem, argument mining, and demonstrated its application across different fields of NLP. However, the task of comparing the similarity of two arguments on the same topic remains an understudied problem.

Previous works (Reimers et al., 2019; Misra et al., 2016) have defined two arguments to be similar if they constitute the same *stance* and cover overlapped *aspects* on the same topic. Consider the two arguments on the topic of net neutrality in Figure 1. Both sentences argue the *pro* side of *net neutrality* and cover an overlapped aspect *prevents monopolies*, and therefore are labeled as *similar*.

Researchers have proposed to compute the argument similarity using averaged contextualized word embeddings and cosine similarity as an unsupervised approach (Devlin et al., 2019; Reimers et al., 2019). However, such representation of arguments may fail to highlight the most salient and representative parts of the arguments that convey their *stances* and *aspects*, as arguments could be redundant and include excessive unnecessary details. For instance in Fig. 1, the underlined parts of the sentences do not contribute any new information needed to decide the *stance* and *aspect* of the arguments. We claim that having these unnecessary information does not help the decision process, and even harms the representations of arguments in an averaged word embeddings setting where the weights of the word embeddings are uniform.

Sent 1: "<> Net neutrality **protects** consumers under near **monopolies** ""Consumers Deserve Protection""."

Sent 2: Net Neutrality **prevents** network providers **from eliminating competing equipment** by making it incompatible with their gateway.

Figure 1: Two similar arguments on the topic *net neutrality*. Bolded parts are the *aspects* of the arguments, and underlined parts are removed by our compression model.

We hypothesize that the *stance* and *aspect* are implied in the parts of the arguments that people find the most *salient*. We believe that this objective aligns with the task of *text summarization*, and propose to use sentence compression as a probe to construct more salient representations for arguments. We experiment our method on the UKP Aspect dataset (Reimers et al., 2019) and demonstrate improvement upon previous state-of-the-art unsupervised methods. We further discuss the advantages and limitations of sentence compression in argumentation tasks with qualitative analysis.

Our contribution of this work is two-fold:

- We propose a simple weighting scheme for salient representations of arguments and im-

prove the SOTA of argument similarity task

- We discuss the benefits and obstacles of using sentence compression in argumentation tasks, and identify a drawback of contextual representaion of arguments.

2 Related Work

2.1 Argument Mining

Argument mining is a closely related task to our target argument similarity task. It is often regarded as the first step of other downstream argumentation tasks like argument similarity (Reimers et al., 2019), argument quality (Wachsmuth et al., 2017), and linking arguments across documents (Cabrio and Villata, 2012). Argument mining has been intensively studied in recent years and demonstrated success given the availability of high-quality, large-scale corpus (Shnarch et al., 2018; Ein-Dor et al., 2019).

For the purpose of this work, we do not focus on the task of argument mining. We assume that all arguments in our experimental dataset are valid arguments, given that they were identified and extracted using well-built argument mining systems.

2.2 Argument Similarity

Comparing to argument mining, argument similarity is an understudied task with few resources available. The task is to compute the similarity score between two sentential arguments. Recent work has defined that two arguments are similar if they constitute the same *stance* and cover the same *aspect* (Reimers et al., 2019). In this work, we follow this convention and regard it as the definition of *similar arguments*.

Most recent unsupervised approach to argument similarity builds on the success of contextualized texts representations (Devlin et al., 2019). For instance, (Reimers et al., 2019) proposes to use averaged word embeddings from pre-trained models like BERT to represent arguments and calculate the cosine similarity between two arguments to decide if they are similar. Other previous works (Misra et al., 2016) also propose to use different features, Ngram, word2vec, etc, to represent arguments. In this work, we follow the approach of (Reimers et al., 2019) and discuss the benefits and obstacles of using contextualized representations and how can we improve them.

There are other supervised methods used to predict the similarity between two sentential arguments. For instance, (Reimers et al., 2019) proposes to finetune a BERT model to predict the similarity between two given arguments. Given that this work focuses on unsupervised methods, we will not discuss the supervised methods used in this literature.

2.3 Sentence Compression

Given a sentence, sentence compression aims to produce a shorter sentence by removing redundant information, while preserving the important and salient content of the original sentence. It is closely related to extractive summarization and deletion-based summarization. A popular benchmark dataset is the Google Sentence compression dataset (Filippova and Altun, 2013), and different neural network based approaches have achieved competent results on it (Lewis et al., 2019; Kamigaito and Okumura, 2020).

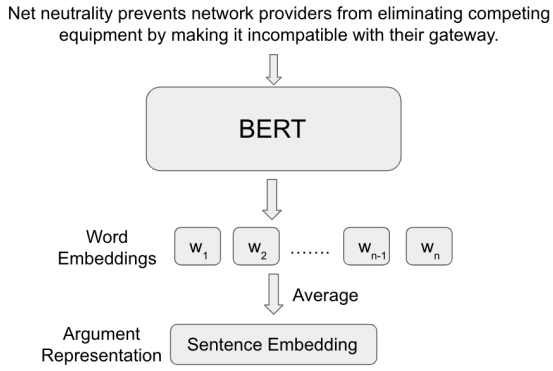
3 Problem Definition

The goal of argument similarity is to define a similarity metric and train a system that takes as input two sentential arguments and returns a scalar value that predicts their similarity.

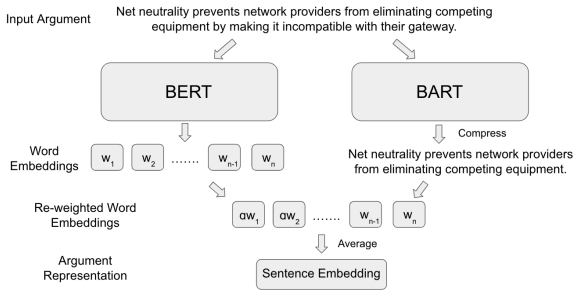
Arguments on controversial topics usually address a limited set of aspects, for example, many arguments on "nuclear energy" address safety concerns. Argument pairs addressing the same aspect should be assigned a high similarity score, and arguments on different aspects a low score (Reimers et al., 2019). Similarly, argument pairs with the same stance on the topic should also be considered as more similar than with the opposite stance (Reimers et al., 2019).

Following the above assumptions and previous work (Reimers et al., 2019; Misra et al., 2016), we define two arguments are similar if they exhibit the same *stance* and *aspect*. Consider the two arguments in Figure 1 on the topic of net neutrality as an example. Both of their *stances* are *supporting* net neutrality, and both of the *aspects* of their arguments are *net neutrality prevents monopolies*. Therefore, these two arguments should be considered *similar*.

To follow the above definitions, we use a dataset that its annotation process aligns with and highlights the aforementioned definitions, the UKP ASPECT dataset (Reimers et al., 2019). We will intro-



(a) Representation of an input argument using averaged word embeddings from pre-trained BERT



(b) Representation of an input argument using re-weighted averaged word embeddings. BART is a summarization model trained on the Google sentence compression dataset.

Figure 2: An example figure of argument representation by (Reimers et al., 2019) (left), and ours (right)

duce the dataset in more details in section 5.1, and the annotation guideline of UKP ASPECT is shown in appendix A.

4 Salient Representation of Arguments via Sentence Compression

We believe that the *stance* and *aspect* are implied in the parts of the arguments that are the most *salient*. However, using only the averaged word embeddings as the representation of arguments (Reimers et al., 2019), the weights of all word embeddings are uniform (figure 2a). That said, the *redundant* parts of the arguments and the *salient* parts of the arguments are equally important in the sentence representation of arguments.

To remedy this problem, we propose to use sentence compression to find the most *salient* parts of the arguments, and re-weight the representation of arguments using this information.

The architecture of our model is shown in figure 2b. Specifically, given an input argument *A*, we compress *A* using a trained compression model BART, and get a compressed sentence *C*. We also pass *A* to a pre-trained BERT model to get its contextualized word embeddings, $w_1, w_2, ..., w_n$. Then for each word embedding w_i, we multiply it by a hyper-parameter α ($\alpha > 1$) to increase its weight if its corresponding word in *A* is kept in the compressed sentence *C*, and keep it the same if the word is removed by the compression model. For instance, in the example shown in figure 2b, the word embeddings of words before *by making* are multiplied by α since they are preserved in the compressed sentence, and word embeddings starting *by making* are kept the same since those words are removed by the compression model. We then take the average of these re-weighted word embeddings as the representation of an argument.

The intuition is that, we want to highlight the parts of the arguments that our compression model finds the most salient, by giving them more weights in the averaged sentence representation. Then in evaluation time, we use cosine similarity to predict the similarity score between two sentence representations, and use a threshold found in validation set to determine if the two arguments are similar.

5 Experiments

5.1 Experiment setup

We use the UKP Aspect dataset as the dataset for our experiments (Reimers et al., 2019). The UKP ASPECT corpus consists of sentences which have been identified as arguments for given topics using an argument mining system. It has 3,595 arguments pairs in 28 topics, and each argument pair is annotated with four degrees of similarity (different topic, no, some, and high similarity). Following (Reimers et al., 2019), we binarize the four labels to only indicate similar and dissimilar argument pairs. Pairs labeled with some and high similarity were labeled as similar, pairs with no similarity and different topic as dissimilar.

We evaluate our methods in the same manner as previous work to reproduce and compare with their results (Reimers et al., 2019). We split the dataset into four folds of testing set with seven topics each fold, and use four other topics as validation set for finding the best threshold and hyperparameter α. Final evaluation results are the average over the four folds and shown in Table 1. We compute

Model	F-mean	F-sim	F-dissim
Reimers et al.	65.64	57.21	74.06
COMP SENT	61.00	48.96	73.04
RE-WEIGHTED	**67.18**	**58.37**	**75.98**

Table 1: Averaged results across all folds. Reimers et al. is our reproduced results of (Reimers et al., 2019) that uses the averaged word embeddings of original arguments as representations for arguments. COMP SENT uses the averaged word embeddings of compressed sentences as the representation of arguments. RE-WEIGHTED is our proposed method that re-weighs the word embeddings of original argument using the compressed sentences. The α used is 1.75.

the marco-average F_{mean} for the F1-scores for the similar-label (F_{sim}) and for the dissimilar label (F_{dissim}). For each pair of arguments, we compute the cosine similarity between their representations, and assign the label similar if it exceeds a threshold, otherwise dissimilar.

5.2 Sentence Compression Model

We use BART (Lewis et al., 2019) as the sentence compression model for our approach. BART is a popular transformer based model that has achieved new state-of-the-art on various summarization tasks (Lewis et al., 2019). We train BART on the Google sentence compression dataset for three epochs (Filippova and Altun, 2013) and use it as our sentence compression model.

5.3 Similarity Metric

Following (Reimers et al., 2019), we use cosine similarity as the similarity metric to compare two representations of arguments. In both models shown in 2a and 2b, given two input arguments, we first generate their sentence representations and then use cosine similarity to predict a similarity score between them. We then use a best threshold found in validation set to determine if the two arguments are similar.

5.4 Results

The average results across all four folds are shown in Table 1, and the F_{mean} scores of each fold is shown in Figure 3. We can see that using our re-weighted representations of arguments effectively improves the results in three folds of the dataset, and the average results across the whole dataset as well.

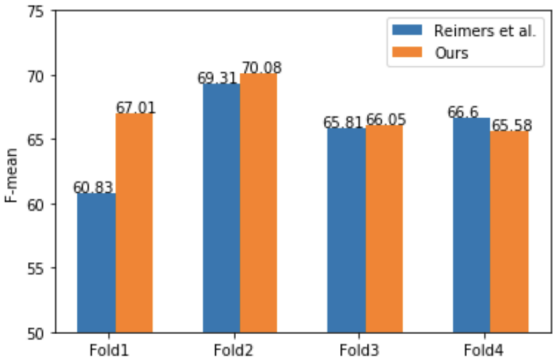


Figure 3: F-mean scores of each fold

5.5 Analysis

How Re-weighted Representation Helps Table 2 shows a pair of arguments that using the original averaged word embeddings, it will be classified as dissimilar (incorrect), while using our re-weighted representation, it will be classified correctly as similar. The bolded part is the part preserved after the compression model. We can see that this observation aligns with our assumption, that the *salient* information preserved by the compression model contains all knowledge we need to determine if the two arguments are similar, i.e. the *stance* and the *aspect*. So highlighting these information by giving them more weights in the word embeddings can help the decision process.

Using only the Compressed Sentence We have also experimented with using the averaged word embeddings of the compressed sentences as the representation of the arguments to compute similarity. The F-score is shown in COMP SENT in Table 1. However, although the compression models preserves most of the salient parts of the argument, it sometimes only preserves the first salient *aspect* and misses the rest. For instance, in the example shown in table 3, the compression model fails to preserve the second aspect of argument 2, potential for fraud, and makes the wrong prediction that they are dissimilar. Therefore, if we only use the embeddings of compressed sentences, it may lose some of the information needed, and makes the performance drop. So using the embeddings of original arguments while expanding the weights of the salient part is a better middle-ground; it alleviates this problem of removing important information, while highlights the most salient part for better prediction.

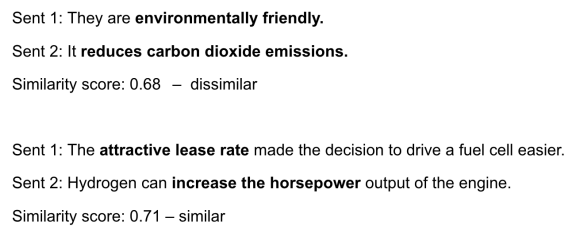


Figure 4: Examples of contextualized embeddings and cosine similarity fail to identify similar arguments. Bolded parts are the *aspects* of the arguments.

Drawbacks of Contextual Embeddings Contextualized word embeddings from pre-trained models like BERT have achieved new state-of-the-art in many NLP tasks like name entity recognition, question answering, and text classification (Devlin et al., 2019; Akbik et al., 2018). However, it may not be the best way to represent arguments. Consider Figure 4 as an example. In the first pair of arguments, both sentences cover the same aspect of *environmentally friendly*, but only receive a cosine similarity score of 0.68. However, in the second pair of arguments, one argues the benefit of hydrogen of attractive rate, and the other argues the benefit of increasing horsepower. Although they cover different aspects, using the word embeddings of these two sentences to compute their cosine similarity gives a similarity score of 0.71. It is still unclear that why contextualized embeddings from BERT and cosine similarity believe that the second pair of arguments is more similar than the first pair, and it indicates that a better representation of arguments other than BERT embeddings is in need.

6 Conclusion

In this work, we propose a re-weighting scheme for argument representation that captures the *salient* parts of the arguments, and show that using our representation of argument outperforms the previous state-of-the-art results in argument similarity task. We further analyze the advantages and obstacles of using sentence compression model to identify the *stances* and *aspects* of arguments, and discuss a potential drawback of using contextualized word embeddings for argumentation tasks.

Argument 1	**It can be easily operated from your smartphone device** by downloading the ARFreeflight application.
Argument 2	Depending on the model, **drones can be controlled remotely** or they may be autonomous, buzzing across the skies, directed by GPS.

Table 2: An example argument pair that are classified incorrectly as *dissimilar* by Reimers et al, and classified correctly as *similar* using our proposed re-weighted representations. The bolded part is the part preserved after the compression model and have embeddings multiplied by α.

Argument 1	Critics' greatest concern about electronic voting machines, however, is that **they might be vulnerable to fraud**.
Argument 2	The public also thinks that **electronic voting machines are prone to unintentional failures** and agreed that they increase the potential for fraud.

Table 3: An example argument pair that using the word embeddings of compressed sentence fails. The bolded part is the part preserved after the compression model.

References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *COLING 2018, 27th International Conference on Computational Linguistics*, pages 1638–1649.

Elena Cabrio and S. Villata. 2012. Natural language arguments: A combined approach. In *ECAI*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Liat Ein-Dor, Eyal Shnarch, Lena Dankin, Alon Halfon, Benjamin Sznajder, Ariel Gera, Carlos Alzate, Martin Gleize, Leshem Choshen, Yufang Hou, Yonatan Bilu, Ranit Aharonov, and Noam Slonim. 2019. Corpus wide argument mining – a working solution.

Katja Filippova and Yasemin Altun. 2013. Overcoming the lack of parallel data in sentence compression. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1481–1491, Seattle, Washington, USA. Association for Computational Linguistics.

Hidetaka Kamigaito and Manabu Okumura. 2020. Syntactically look-ahead attention network for sentence compression.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training

for natural language generation, translation, and comprehension.

Amita Misra, Brian Ecker, and Marilyn Walker. 2016. Measuring the similarity of sentential arguments in dialogue. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 276–287, Los Angeles. Association for Computational Linguistics.

Nils Reimers, Benjamin Schiller, Tilman Beck, Johannes Daxenberger, Christian Stab, and Iryna Gurevych. 2019. Classification and clustering of arguments with contextualized word embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 567–578, Florence, Italy. Association for Computational Linguistics.

Eyal Shnarch, Carlos Alzate, Lena Dankin, Martin Gleize, Yufang Hou, Leshem Choshen, Ranit Aharonov, and Noam Slonim. 2018. 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)*, pages 599–605, Melbourne, Australia. Association for Computational Linguistics.

Christian Stab and Iryna Gurevych. 2014. Identifying argumentative discourse structures in persuasive essays. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 46–56, Doha, Qatar. Association for Computational Linguistics.

Henning Wachsmuth, Nona Naderi, Ivan Habernal, Yufang Hou, Graeme Hirst, Iryna Gurevych, and Benno Stein. 2017. Argumentation quality assessment: Theory vs. practice. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 250–255, Vancouver, Canada. Association for Computational Linguistics.

A Dataset Annotation

Read each of the following sentence pairs and indicate whether they argue about the same aspect with respect to the given topic (given as "Topic Name" on top of the HIT). There are **four options**, of which one needs to be assigned to each pair of sentences (arguments). Please read the following for more details.

- **Different Topic/Can't decide**: Either one or both of the sentences belong to a topic different than the given one, or you can't understand one or both sentences. If you choose this option, you need to very briefly explain, why you chose it (e.g. "The second sentence is not grammatical", "The first sentence is from a different topic" etc.). For example,

  Argument A: "*I do believe in the death penalty, tit for tat*".

  Argument B: "*Marriage is already a civil right everyone has, so like anyone you have it too*".

- **No Similarity**: The two arguments belong to the same topic, but they don't show any similarity, i.e. they speak about completely different aspects of the topic. For example,

  Argument A: "*If murder is wrong then so is the death penalty*".

  Argument B: "*The death penalty is an inappropriate way to work against criminal activity*".

- **Some Similarity**: The two arguments belong to the same topic, showing semantic similarity on a few aspects, but the central message is rather different, or one argument is way less specific than the other. For example,

  Argument A: "*The death penalty should be applied only in very extreme cases, such as when someone commands genocide*".

  Argument B: "*An eye for an eye: He who kills someone else should face capital punishment by the law*".

- **High Similarity**: The two arguments belong to the same topic, and they speak about the same aspect, e.g. using different words. For example, Argument A: "*An ideal judiciary system would not sentence innocent people*".

  Argument B: "*The notion that guiltless people may be sentenced is indeed a judicial system problem*".

Your rating should not be affected by whether the sentences attack (e.g. "*Animal testing is cruel and inhumane*" for the topic "*Animal testing*") or support (e.g. "*Animals do not have rights, therefore animal testing is fair*" for the topic "*Animal testing*") the topic, but only by the aspect they are using to support or attack the topic.

Figure 5: The annotation guidelines of UKP Aspect dataset