Validating AMR-based Argument Similarity on Novel Corpora

Atanasoska Tamara, and **Ryazanskaya Galina,** and **Emanuele De Rossi**
University of Potsdam

Abstract

This paper presents a replication, validation, and extension of a study in Abstract Meaning Representation (AMR) based argument similarity assessment (Opitz et al., 2021). We successfully replicate the AMR argument similarity metric suggested in the original paper and validate the results on two additional argument similarity corpora (AFS and BWS (Misra et al., 2016; Thakur et al., 2020a)). Additionally, we experiment with the conclusion generation process suggested in the original paper and find that pre-trained summarization performs better than conclusion generation, probably due to fine-tuning instability. The analysis of the effects of argument length on metric performance reveals that predicting argument similarity with the AMR metric is easier on shorter arguments. The metric showed state-of-the-art or comparable performance on all three corpora (f1 scores of 67.84% on UKP, 65.34% in AFS, and 63.77% on BWS).

1 Introduction

Argument similarity is the task of determining how similar two arguments are. Arguments can be defined as text fragments that expose one's opinion on a certain topic, and they can be usually divided into the claims and the premises that support those claims (Lenz et al., 2019). Assessing argument similarity in a comprehensible and clear way is no easy task since it is difficult to determine the set of features shared between two arguments that increase or decrease the similarity between them (Opitz et al., 2021).

Finding a way to meaningfully assess argument similarity could bring improvements in many argument-related tasks, such as argument clustering (Reimers et al., 2019), counter-argument retrieval (Wachsmuth et al., 2018), argument search (Chesnevar and Maguitman, 2004; Ajjour et al., 2019), and intelligent tutoring (Bai and Stede, 2022).

The present paper presents a validation and extension of Opitz et al. (2021), where the authors use Abstract Meaning Representation (AMR) graphs and apply graph similarity metrics to evaluate the similarity between the arguments. AMR graphs are trees that represent the semantic structure of a sentence. The motivation behind using AMR graphs for this approach is that AMR graphs can capture the meaning of a sentence, independently from its syntactic structure, in a machine-readable format.

The authors also automatically generate conclusions from the arguments to test two hypotheses: that similar arguments lead to similar conclusions, and that extending arguments with automatically inferred conclusions could improve the argument similarity task performance.

Our contributions consist in (1) reproducing the results of Opitz et al. (2021), (2) validating the AMR argument similarity metric on two novel corpora, AFS and BWS (Misra et al., 2016; Thakur et al., 2020a), both with binary classification and with correlation analysis, (3) comparing the performance of summarization and conclusion generation for the AMR argument similarity metric, and (4) exploring the relationship between the performance of the AMR argument similarity metric and the argument length.

We find that:

- the results reported by Opitz et al. (2021) were generally reproducible, though insufficiently documented;
- similar results hold on the other argument similarity corpora, with stronger patterns apparent with correlation analysis of the AMR argument similarity metric;
- summarization performed significantly better than conclusion generation for argument similarity prediction, probably due to low conclusion quality that resulted from model insta-

bility and insufficient documentation of the original fine-tuning process;

- argument length has negative effects on the performance of the AMR argument similarity metric, which could be bound to lower performance of some of the pipeline components on longer texts (e.g., of the AMR parser or the graph similarity assessment algorithm).

Acknowledging the importance of reproducibility, we deem it necessary to provide the documented code that we used for the present project [1]. The repository follows the code provided in three repositories: *amrlib*[2] (Cai and Lam, 2020), *amr-metric-suite*[3] (Opitz et al., 2020), and *amr-argument-sim*[4] (Opitz et al., 2021). The code from these repositories features minor alterations as well as additional files created to (1) fill in for the missing code based on the descriptions provided by the authors, (2) generalize the method to our extended tasks based on the existing code, (3) analyze the results by correlation and by length. Additionally, our repository includes extended documentation to improve usability.

2 Related Work

As the argument similarity task requires comparing the arguments, the assessment is carried out on corpora consisting of pairs of arguments with manually annotated argument similarity scores. In this section, we overview the argument similarity corpora (2.1) as well as the different approaches to assessing argument similarity (2.2). A separate section is dedicated to the paper being replicated, Opitz et al. (2021) (2.4), and to the methods used in it, namely, abstract meaning representation (2.3) and the conclusion generation (2.5).

2.1 Argument Similarity Corpora

We are aware of three freely available argument similarity corpora: UKP Aspect Corpus (Daxenberger et al., 2019; Reimers et al., 2019), BWS Argument Similarity Corpus (Thakur et al., 2020a,b), and Argument Facet Similarity Corpus (Misra et al., 2016, 2017). The information on the corpora is summarized in Table 1[5].

It is important to note that the similarity scores differ between the corpora and are not quite directly comparable. The UPK scoring system from 1 to 4 is not a scale: 1 being a different topic, 2 - not similar, 3 - somewhat similar, and 4 - very similar. For BWS the similarity score between 0 and 1 was inferred from preference judgements, not direct similarity questions. As for AFS corpus, the 1 - 5 scores are also not a direct scale of similarity. The score descriptions were as follows:
(5) Completely equivalent, mean pretty much exactly the same thing, using different words.
(4) Mostly equivalent, but some unimportant details differ. One argument may be more specific than another or include a relatively unimportant extra fact.
(3) Roughly equivalent, but some important information differs or is missing. This includes cases where the argument is about the same ***FACET*** *but the authors have different stances on that facet.*
(2) Not equivalent, but share some details. For example, talking about the same entities but making different arguments (different facets).
(1) Not equivalent, but are on the same topic.
(0) On a different topic.

A facet was defined as "a low-level issue that often reoccurs in many arguments in support of the author's stance or in attacking the other author's position". [6]

2.2 Approaches to the Argument Similarity Task

There is a variety of approaches to the argument similarity tasks, ranging from predicting similarity based on a set of features (Misra et al., 2016, 2017; Bai and Stede, 2022) to neural network-based methods (Thakur et al., 2020b; Behrendt and Harmeling,

[1] https://github.com/TamaraAtanasoska/AMR_ArgumentSimilarity

[2] https://github.com/bjascob/amrlib

[3] https://github.com/flipz357/amr-metric-suite

[4] https://github.com/Heidelberg-NLP/amr-argument-sim

[5] Throughout the paper, the correlation and f1 scores are multiplied by 100 for higher resolution.

[6] The authors specify that "*There are many ways to argue for your stance on a topic. For example, in a discussion about the death penalty you may argue in favor of it by claiming that it deters crime. Alternatively, you may argue in favor of the death penalty because it gives victims of the crimes closure. On the other hand, you may argue against the death penalty because some innocent people will be wrongfully executed or because it is a cruel and unusual punishment. Each of these specific points is a facet. For two utterances to be about the same facet, it is not necessary that the authors have the same belief toward the facet. For example, one author may believe that the death penalty is a cruel and unusual punishment while the other one attacks that position. However, in order to attack that position they must be discussing the same facet.*"

	pairs	topics	scale	inter-rater agreement	argument length
UKP	3595	28	1-4 (discrete)	0,43 (Krippendorff's α)	136
AFS	6000	3	0-5 (continuous)	68 (Pearson correlation)	150
BWS	3400	8	0-1 (continuous)	66 (Spearman correlation)	153

Table 1: Comparative description of the arguments similarity corpora, including the number of argument pairs and topics of each argument similarity corpus, the similarity scale, the reported inter-rater agreement, and the average argument length in words.

2021; Bai and Stede, 2022) and graph similarity-based approaches (Lenz et al., 2019; Bergmann et al., 2019; Opitz et al., 2021).

2.2.1 Feature-Based Approaches

Misra et al. (2016, 2017) use curated features to predict the similarity of the arguments. The features included simple features like n-gram overlap, as well as features obtained from various pre-trained linguistic tools, such as textual similarity score produced by UMBC Semantic Similarity tool, distributional similarity produced by DISCO tool, as well as Linguistics Inquiry Word Count (LIWC) tool and ROUGE metric. The authors used SVM and linear classifier on top of these features to predict the similarity of the arguments on AFS corpus. The authors report the best correlation coefficients of 53.2 and 54.0 for these models respectively.

2.2.2 Embedding-Based Approaches

The argument similarity task is somewhat similar to another classical NLP task called semantic textual similarity (STS). STS is a supervised task that consists in predicting the semantic similarity of a pair of sentences (Šarić et al., 2012). As with argument similarity, STS models often rely on a set of different text comparison features (including sentence embedding similarity, syntactic roles similarity, syntactic dependency overlap, etc.) to calculate a similarity score between two input sentences (Šarić et al., 2012). STS has been an important tool in multitask training of large models, such as BERT (Devlin et al., 2018).

Reimers et al. (2019) report that fine-tuned BERT showed a 74.01 F1-score on UKP corpus, outperforming simpler models such as TF-IDF (61.12), InferSent (66.21 and 64.94), ELMo (64.47), and BERT embeddings (65.39). They report the human performance on the task to be 78.34.

Reimers and Gurevych (2019) present a model, called SentenceBERT (SBERT), which is a network that fine-tunes BERT using siamese and triplet network structures to derive semantically meaningful sentence embeddings of the input sentence. Those embeddings can be then compared using cosine-similarity to determine how similar a pair of sentences is.

Importantly, there are some key differences between the semantic similarity and argument similarity tasks, as argument similarity requires that the logic of the arguments is similar, though the details may differ. Thus, while SBERT performs well on tasks like STS and paraphrase identification, it gives worse results when assessing the similarity of arguments (Behrendt and Harmeling, 2021). Still, models that perform well on STS can serve as a basis for the argument similarity task. Several papers modify SBERT to improve its performance on the argument similarity task, including Augmented SBERT (Thakur et al., 2020b) and ArgueBERT (Behrendt and Harmeling, 2021).

Augmented SBERT is a data augmentation approach proposed in Thakur et al. (2020b) to improve the performance of the SBERT bi-encoder. The approach, when evaluated for the argument similarity task on BWS corpus, presented an 8 percent improvement with respect to the SBERT baseline (61.20), showing 69.76 on in-topic argument similarity.

ArgueBERT is a model fine-tuned in a siamese SBERT architecture, but it is pre-trained using three argument-related tasks: argument similarity prediction, argument order prediction, and argument graph edge validation (Behrendt and Harmeling, 2021). The model is then evaluated using the three argument similarity corpora described above: UKP, AFS, and BWS (Daxenberger et al., 2019; Misra et al., 2017; Thakur et al., 2020a), along with a paraphrase dataset named MRPC (Microsoft Research Paraphrase Corpus, Dolan and Brockett (2005)). The correlation analysis showed improvements on the argument similarity corpora compared to the SBERT model: 35.33 vs 32.04 on UKP, 38.25 vs 38.02 on AFS, and 43.44 vs 38.55 on BWS evaluated with Spearman rank correlation.

Bai and Stede (2022) explore using argument similarity scoring for a tutoring system that compares answers written by students on questions about a source text against a reference answer. The system then assesses the similarity level between the student's answer and the reference answer and presents concepts or ideas that the student might have missed. The paper introduces a novel German argument similarity corpus with an agreement level of 83.5 and assesses several approaches to argument similarity scoring for the German language, including LSA cosine similarity, SVM and random forest over lemmatized unigram and n-gram overlap as well as LSA vectors and cosine similarity, and a fine-tuned pre-trained German BERT model. The BERT model showed the best result with a Spearman correlation score of 77.3.

2.2.3 Graph-Based Approaches

As arguments often have a complex logical structure, many approaches represent them with graphs that capture this structure. Bergmann et al. (2019) and Lenz et al. (2019) try to compare such argument graphs in Argument Interchange Format to assess their similarity. They apply the methods of semantic textual similarity to the graph nodes, which are, in their use case, longer text fragments, such as sentences. They use word and sentence embeddings such as word2vec as well as fine-tuned models such as BiLSTM and USE and experiment with combining embeddings. They assess the model on 112 manually annotated arguments and show that USE is able to achieve 71.3 precision (with human performance at 71.7 precision).

Opitz et al. (2021), on the other hand, use graphs on the level of individual sentences to assess the similarity of arguments on UKP corpus. The graphs used are abstract meaning representation graphs, covered in the following section (2.3), and they have single-word nodes. As we reproduce and validate the method suggested in this paper, we provide a detailed description of the original work in section 2.4.

2.3 Abstract Meaning Representation (AMR)

Abstract Meaning Representation (AMR), developed by Langkilde and Knight (1998), is a method of semantic representation, that represents sentences as acyclic directed graphs with a single root (Flanigan et al., 2014). Figure 1 demonstrates an example of an AMR graph. In AMR graphs, the nodes represent concepts (such as "boy" or "go" in the example), and the labeled directed edges represent relationships between the nodes.

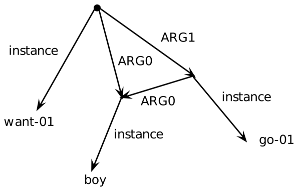

Figure 1: AMR graph for the sentence *The boy wants to go* (Cai and Knight, 2013).

AMR graphs are designed in such a way that they can represent a sentence independently from its syntactic structure (i.e., two sentences formulated very differently but with a similar meaning can be represented with similar AMR graphs (Banarescu et al., 2013). The idea underlying this approach is that by extracting the semantic elements of a sentence along with the relationship between those elements, one can extract the intrinsic *meaning* of that sentence more efficiently (Dridan and Oepen, 2011).

There is a variety of AMR parser models, and a comparison of some of the state-of-the-art ones can be found at *papers with code* website [7]. The AMR parser that we adopted for this project was the one used by Opitz et al. (2021), namely, the *amrlib* parser (Cai and Lam, 2020).

2.4 Opitz et al. (2021)

Opitz et al. (2021) introduce an argument similarity metric based on AMR graphs. As the AMR graph has both concept nodes and relation edges, the authors assess the importance of these two graph components for the argument similarity task by introducing weighting schemes. One, named *standard*, combines the nodes and the edges similarity when comparing two argument AMR graphs. The *concept* and *structure* weighting schemes give 3 times more weight to nodes and edges respectively when comparing two argument AMR graphs. This approach is further extended by estimating the similarity of automatically generated conclusions in the same manner.

The authors test the metric on UKP corpus modified into a binary prediction problem (*highly similar / somewhat similar* vs *not similar / different topic*). They generate a conclusion for each argument using a T5 model pre-trained on the sum-

[7] `https://paperswithcode.com/task/amr-parsing/`

marization task and fine-tuned on a modified persuasive essay corpus to turn the summarization into conclusion generation (Stab and Gurevych, 2017). Both arguments and conclusions are represented with AMR graphs obtained using *amrlib* parser (Cai and Lam, 2020). The AMR graphs are compared using S2MATCH graph similarity metric (Opitz et al., 2020), which is a modified version of the SMATCH metric from Cai and Knight (2013). The weighting schemes variations of it mentioned above are applied to the S2MATCH. The authors calculate all the variants of S2MATCH for the AMR graphs of both arguments and generated conclusions, and combine the scores, giving the majority of the weight to the argument. The resulting similarity score is used for the binary prediction. The threshold for binary similarity prediction is selected via topic-wise cross-validation.

The results show strong evidence for the hypothesis that AMR graphs can be useful for assessing argument similarity, especially when the AMR metrics are adapted to give more importance to concrete concepts (concept weighting: 68,7 f1). There is only weak evidence for the hypothesis that similar arguments lead to similar conclusions: the metrics only slightly improve when the AMR graphs of the generated conclusions are added (less than 1% improvement), and the conclusions used without the arguments perform much worse (60,29 with concept weighting).

Beyond introducing the new AMR metric for assessing argument similarity, the authors explore the generated conclusions for conclusion quality and usefulness, but this aspect of the original experiment lies beyond our replication task, which is concentrated on the proposed argument similarity metric.

2.5 Conclusion Generation

Conclusion generation is a generative task that is aimed at generating a textual conclusion given a premise or a list of premises. The inferential link between the premises and the conclusion lies at the core of an argument (Heinisch et al., 2022).

2.5.1 Conclusion Generation as an Open Research Question

As a part of reproducing Opitz et al. (2021), we needed to train the conclusion generation model. As mentioned above, the model they used was based on a pre-trained summarization model which was fine-tuned on the persuasive essay corpus (Stab and Gurevych, 2017) to produce conclusions, or rather to make the generated summaries more "conclusion-like". However, the conclusion generation approach used in this paper has been abandoned by the researchers, in favor of the most recent approaches, like the ones discussed in Heinisch et al. (2022).

Conclusion generation and especially argumentative conclusion generation is still an open research question. Many factors contribute to its complexity, most notably the fact that many different conclusions can be deemed valid for any given argument Heinisch et al. (2022). Further, as Syed et al. (2021) put it, "*conclusions [...] typically depart significantly from the argumentative text they are derived from, paraphrasing it, and more than half the time abstracting over it. Authors typically tailor their conclusions to the occasion; and in many cases, they are not necessarily made explicit.*" This makes the pool of acceptable conclusions vast, and if the most important conclusion is implicit, it may be very difficult for a model to generate it. Another issue arises from the validity of the factual assertions which is often challenging to determine (see Gretz et al. (2020)).

The texts that the conclusion generation models produce are also impacted by the more general issues that affect any natural language generation(NLG) tasks. The "text-to-text" generative models are mapping input text to output text. However, the input text does not provide a full insight into the intended meaning. There are efforts aimed at incorporating the internal knowledge hidden in the input text and the external knowledge found in knowledge bases and knowledge graphs (Yu et al., 2022), echoing the decade-old research aimed at incorporating more real-world context, such as the so-called commonsense knowledge, into any NLP task (see for example Speer et al. (2017)).

One prominent conclusion generation approach which uses BERT embeddings and focuses on the "sufficiency" aspect of the conclusions is presented in Gurcke et al. (2021). The authors use several pretrained transformer-based language models both unsupervised and supervised to generate conclusions. The paper is not only aimed at generating conclusions but also trying to see if they make logical sense. The main contribution of the paper lies in the second step, where the authors try to asses the "sufficiency" of a generated argument or in other words try to determine whether "the premises in the argument are worth drawing the conclusion from". A sufficient argument is defined as an argument the premises of which rationally support it. They do the sufficiency assessment using two approaches. The first, called *direct sufficiency assessment*, uses RoBERTa, a transformer-based model to perform the assessment. The second, called *indirect sufficiency assessment*, incorporates structural knowledge together with the conclusions for the assessment. [8]

2.5.2 Evaluating Generative Models

Another hindering aspect for any generative NLP models including conclusion generation is that there are still currently no efficient ways of evaluating the generation quality. Currently, the gold standard is still qualitative evaluation done by humans employing various strategies van der Lee et al. (2019) and sometimes hybrid approaches with machine scores like BLEU. Yet, the metrics like BLEU are considered poor substitutes for human evaluation (Sellam et al., 2020), and many competing evaluation frameworks are appearing to mitigate this problem. The manual evaluation was the evaluation method carried out by (Opitz et al., 2021) in the second part of their paper.

This complexity of evaluation makes training dependent on human feedback, and in our own experience that, in turn, makes the research slower and more resource intensive in terms of human supervision.

2.5.3 Conclusion Generation and Argument Length

One extension that we carried out concerned the effect that the length of the argument had on the performance of the metric. The motivation behind this extension was that the model that is pre-trained on summarization could be sensitive to lengths. A summarization is usually a task applied to a longer text to obtain a succinct version of it. Thus, the conclusion generation from a summarization-based model could potentially be more meaningful on longer premises or in cases where there are multiple premises. However, the effect of length on the final similarity metric performance could also depend on the performance of other components, such as the AMR parser and the S2MATCH algorithm, which could also be sensitive to length.

3 Problem Statement

Our project aims to shed more light on the two hypotheses postulated by Opitz et al. (2021) concerning the usefulness of the AMR graphs and of automatically generated conclusions for the argument similarity task. Beyond reproducing the results of Opitz et al. (2021), we validate and extend them in three ways:

1. we replicate the results on two other argument similarity corpora. Opitz et al. (2021) experiment with only one corpus, namely, UKP (Daxenberger et al., 2019), and we assess the reproducibility of the results on two other argument similarity corpora: AFS and BWS (Misra et al., 2016; Thakur et al., 2020a).

2. we compare how well the T5-based summarization model suggested by the authors for conclusion generation performs before and after fine-tuning on the modified Persuasive Essays Corpus, comparing the usefulness of summaries and conclusions.

3. we explore how the length of the argument affects the conclusion generation and the metric performance overall.

The project workflow naturally required similar data processing across all the corpora. Figure 2 gives an overview of the models and data in our project. Each dataset consisted of argument pairs and corresponding similarity scores. The scores had to be transformed to binary labels and, for the two datasets that allowed for it (AFS & BWS), also re-scaled to a 0-1 scale for correlation analysis.

We replicated the way the original paper (Opitz et al., 2021) automatically generated the conclusions from the arguments using the T5 model (Raffel et al., 2019) pre-trained on summarization tasks and fine-tuned on a modified version Persuasive Essays Corpus (Stab and Gurevych, 2017), which contains premise-conclusion samples. The pre-trained model was used to obtain summaries and its fine-tuned version was used for conclusion generation. Both summarization and conclusion generation are described in section 4.3.

Aside from adjusting different scoring schemes, the data processing was identical for the three cor-

[8] Initially, we wanted to compare this approach to the one used in the paper. Unfortunately, crucial data was missing from the project GitHub that we needed to replicate this model. At our request, the data was provided by the researchers. However, due to computational and time restrictions, we had to leave out the comparison between these two models, and only concentrate on the approach used by Opitz et al. (2021).

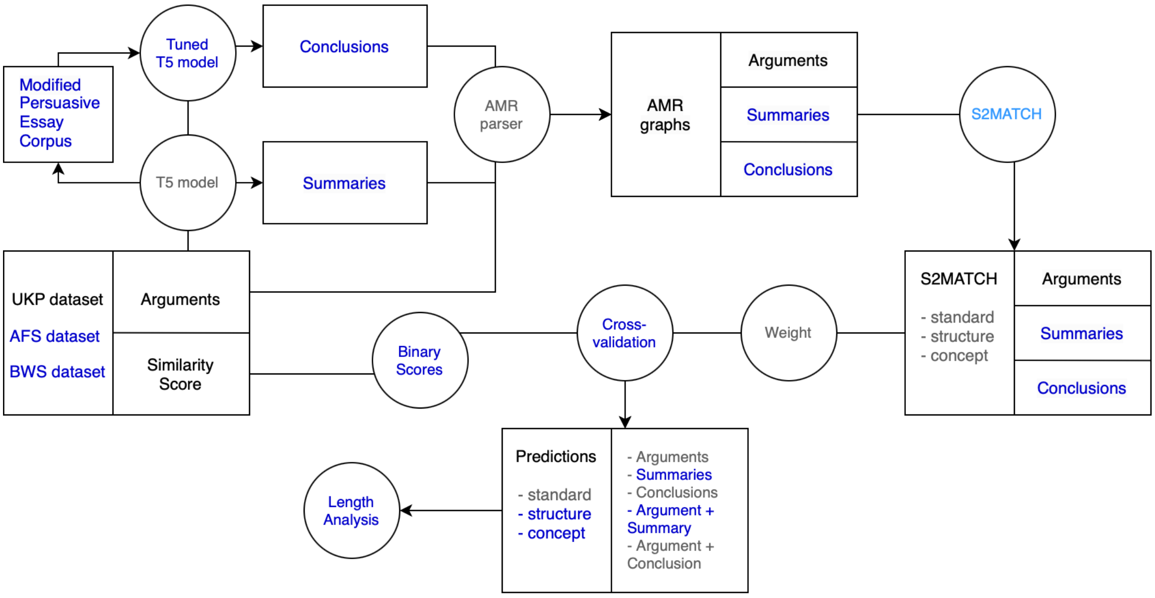

Figure 2: The use of data and models in our project. The square blocks represent corpora and textual data. The round blocks represent models and processing. The dark blue inscriptions represent the extensions. The parts in black were adopted directly from (Opitz et al., 2021). The grey represents other pre-trained resources. Light blue represents the modifications to the original paper that we had to introduce.

pora. The summaries and conclusions were generated for each argument. Then, the arguments, summaries, and conclusions were all put through the AMR parser model to obtain AMR graphs. The pairs of AMR graphs for the pairs of arguments, as well as the pairs of their summary and conclusion graphs, were compared using S2MATCH metric. S2MATCH weighted variants *standard*, *concept*, and *structure* were applied to obtain the similarity scores. The arguments, summaries, and conclusions were evaluated by predicting the binary label and by correlation with the true similarity score. They were assessed separately (argument, summary, conclusion) and in combination (argument + summary and argument + conclusions with a weight of 0.95 given to the argument). A threshold for each corpus was selected using topic-wise cross-validation. Each corpus was also binned by length to analyze the effect of length on prediction quality.

4 Models

As shown in figure 2, there were four separate models used for the project: AMR parser (section 4.1), S2MATCH (the AMR graph similarity metric introduced in (Opitz et al. (2020), section 4.2; and the summarization model along with its fine-tuned conclusion generation version (section 4.3). The models are described in the following sections.

4.1 AMR parsing

We used a pre-trained model named amrlib (Cai and Lam, 2020) for AMR parsing as it was the model used by (Opitz et al., 2021). This parser is based on a pre-trained T5 model. Although there are many newer pre-trained models available, to be able to get as close as possible results to the paper, we picked the oldest model, which was uploaded around the time of the paper (Opitz et al., 2021). The base model was not documented in the original article, and as we wanted to see how the results would compare to the original ones as well as to performance on other corpora, we wanted to keep the models used as close to the original paper as possible.

4.2 S2MATCH

The algorithm for calculating the AMR graph similarity scores is at the core of the paper we replicated (Opitz et al., 2021). It goes through the possible matches of the graph nodes and the graph edges taking into account the cosine similarity between the pair of concepts at the nodes, and the match or mismatch of the relation between them. The algorithm keeps track of the best possible match, that is the one with the most overlap between both concepts in the nodes and the relations between them. The best alignment of the graphs is used as the final similarity score.

The paper describes standard S2MATCH similarity score (Opitz et al., 2020), called *standard*, and the two weighted variations of it *concept* and *structure* (Opitz et al., 2021). These two variations differ from each other for the fact that *concept* focuses on the conceptual overlap between sentences, by putting a triple weight on concept matching (i.e., the nodes of the graphs), while *structure* focuses on the structural overlap, by putting a triple weight on relation matches (i.e., the edges of the graphs). The *standard* keeps the two graph components weighted equally. As the code for calculating the modified weighting schemes was missing, we replicated it based on the description provided by the author and validated it by comparing our scores to the ones published by the authors on the original AMR graphs of UKP corpus.[9]

We used the two repositories for generating the AMR graphs and calculating the S2MATCH scores: the original repository to calculate the S2match score[10] and the GitHub repository of Opitz et al. (2021)[11].

The S2MATCH scores were generated separately for the pairs of arguments, pairs of summaries, and pairs of conclusions. Then, to obtain the combined scores, the S2MATCH of the argument was combined with a weight of 0.95[12] with the S2MATCH of either summary or conclusion.

4.3 Summarization and Conclusion Generation

4.3.1 Summarization

The summarization is performed by the T5 model Raffel et al. (2019), which is already pre-trained for the summarization task. Here, we replicated closely what the authors described in the original paper.

4.3.2 Fine-Tuning Replication Starting Point

The authors of the original paper did not publish or save any inference-ready model or code for the summary and conclusion generation. Upon request, they provided us with some information on the way the fine-tuning was carried out, which we relied on for our fine-tuning. The details can be found in Appendix A. The two points where we had to deviate from these descriptions are described below (4.4.1).

4.4 Fine-Tuning Dataset

Opitz et al. (2021) states that Persuasive Essay Corpus (Stab and Gurevych, 2017) was modified to be used for fine-tuning the summarization model. Persuasive Essay Corpus consists of 402 persuasive essays annotated for argumentative structure with major claims, claims supporting or contradicting them, and premises supporting or contradicting the claims or each other. For fine-tuning, all claims with their annotated premises were retrieved from all annotated premise-conclusion-pairs in this corpus. All annotated major claims with their supportive claims were also used as premise-conclusion pairs. Multiple premises or supportive claims of a single claim were concatenated separated by '.' in document order. The authors of the original paper did not mention that there are also cases where premises support other premises. We decided to treat them as normal premise-conclusion pairs. All the resulting pairs of concatenated premises and their conclusions were used as the text to summarize and the "summary", to turn the summarization model into a conclusion generation model.

4.4.1 Fine-Tuning the Summarization Model for Conclusion Generation

There was no fine-tuning code available, so we used a notebook listed on the profile site for T5 within the HuggingFace hub[14] as a base for our fine-tuning. The decision to use a template online was based on the idea to keep everything as general as possible. We applied the same idea when it comes to using the starting point information in the section above. We used both the information about the training parameters as well as the information about the inference parameters in full, with the respective differences that needed to be applied to use them within the PyTorch framework. One more prominent difference is that we chose to use the `t5-base` model instead of `t5-small` used by the authors (A), as the `t5-small` produced very poor results dissimilar to ones shown in the original paper. Another difference was that we had

[9] https://github.com/TamaraAtanasoska/AMR_ArgumentSimilarity/blob/main/repro_repos/amr-metric-suite/py3-Smatch-and-S2match/smatch/s2match.py

[10] https://github.com/flipz357/amr-metric-suite

[11] https://github.com/Heidelberg-NLP/amr-argument-sim

[12] suggested in the original paper (Opitz et al., 2021).

[14] https://colab.research.google.com/github/abhimishra91/transformers-tutorials/blob/master/transformers_summarization_wandb.ipynb#scrollTo=OKRpFvYhBauC

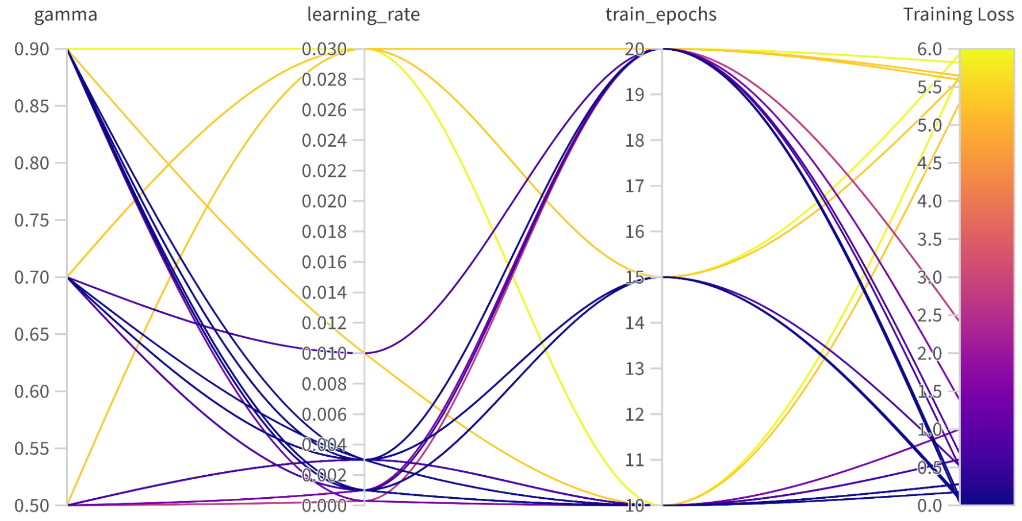

Figure 3: A graph showing the combination of parameters attempted during the hyperparameter search ("sweep") on the W&B platform including gamma value for the exponential learning rate, learning rate, and train epochs in relation to the training loss. The full details of the sweep can be accessed through the public link[13].

to use `sentencepiece` instead of `wordpiece` which was suggested by the authors (A). This is because the T5 model we chose is trained using `sentencepiece` and we decided that we have to use compliant tokenization.

To find the best parameters for the model, we used the Weights & Biases platform and performed a so-called "sweep", which allowed us to tune the chosen hyperparameters for given ranges. A chart featuring all the 30 runs of the sweep can be found in figure 3. We used a 90-10 train-dev split following the original paper.

It was not possible to search for optimal batch size as the University GPU server would only allow for a batch size of 4 as the maximum value. The sweep analysis showed that from the few parameters we tuned, namely, learning rate, epochs, the gamma value for the ExponentialLR[15], the learning rate had the overwhelming effect on the training loss.

We made the best fine-tuned model available in private cloud storage[16]. The parameters for the best model were: `conclusion_len` 128, `gamma` 0.9, `learning_rate` 0.003, `max_len` 512, `seed` 42, `train_batch_size` 4, `train_epochs` 20, `valid_batch_size` 4, `valid_epochs` 1. The model converged to 0.03 training loss.

4.5 Evaluation

The performances of the AMR argument similarity metric and its variants were evaluated in two ways.

First, binary similarity scores were created for all the corpora. UKP was separated into bins of similarity scores of 1-2 vs 3-4, BWS was split at a similarity score of 0.5, and UKP at 3. The similarity metric was evaluated at being able to predict these binary similarity scores. The threshold was selected using topic-wise cross-validation (4 folds on UKP and BWS, and 3 folds on AFS). The resulting thresholds were also used for predictions for the length analysis.

Second, the similarity scores of the two corpora that provided continuous scores (AFS and BWS) were rescaled to [0,1] interval using min-max normalization. The similarity scores predicted by the AMR argument similarity metric were evaluated using Spearman correlation, with higher correlation scores indicating better similarity assessment.

5 Results

Table 2 summarizes the binary prediction results across the three corpora and compares them to the results reported by Opitz et al. (2021). These scores highlight the contribution of two main metric fac-

[15] the search included:
learning rates of 0.1, 0.003, 0.001, 0.0003, 0.0001,
attempted numbers of epochs of 10, 15, 20
gamma values of 1.0, 0.9, 0.7, 0.5, 0.3

[16] https://drive.google.com/file/d/1X0g7T5lZ0UVzNFPA4dZ7WOzKthUVtwXa/view

	original-UKP	**UKP**	**AFS**	**BWS**	**mean**
argument only	64,78	65,31	**63,11**	61,61	63,34
conclusion only	58,03	50,79	53,72	52,50	52,34
summary only		60,95	57,87	59,68	59,50
argument + conclusion	**65,35**	65,02	63,10	61,71	63,27
argument + summary		**65,60**	63,01	**61,73**	**63,45**
standard	62,99	62,24	60,56	59,45	60,75
concept	**65,72**	**63,38**	**61,68**	**60,92**	**61,99**
structure	59,45	58,98	58,24	57,96	58,39
mean	62,72	61,53	60,16	59,44	60,38
stdev	4,55	6,22	4,25	3,94	4,81

Table 2: f1 score summarized across model aspects. The Original-UKP reports the results from Opitz et al. (2021). The first part compares the results using argument / summary / conclusion on their own or combined with a weight of 0.95 given to the argument. The second part compares the weighting schemes: standard, concept, and structure suggested in Opitz et al. (2021). The last part summarizes the results for each corpus across metric variants. The mean is calculated across our replication results, and it does not include the Original-UKP column. The maximal values of each corpus are highlighted in bold within aspects.

tors: the use of the argument, conclusion, or summary; and the weighting scheme (standard, concept, or structure). These are obtained by averaging across the other factor. A more detailed table, that includes the results for all factor combinations can be found in the appendix B (table 4). The following sections discuss these results.

5.1 Replication Results

As can be seen in table 2, the results of Opitz et al. (2021) are generally replicated (within 1-2%), except for the low performance of the conclusions used on their own that we observed. The performance of the metric that we observed was on average 1% lower, possibly due to the large differences in the conclusion quality. Interestingly, the argument-only metric performed better for us than for the authors (by 1%) and even slightly outperformed the combination with the conclusion.

There were a few differences between our results and the ones reported by Opitz et al. (2021). From the table 4 in appendix B, one can see that the best metric was using argument only with concept weight (68%) followed by concept on the argument with its conclusion (67%), while the authors observed the inverse pattern. Secondly, we had lower-quality conclusions, so the results on them are around 1% lower when combined with the argument and 6-10% lower when used on their own. The reasons for lower conclusion quality may lie in model instability, its high dependence on fine-tuning data being similar to the training data, or parameter incompatibility, which is discussed further in section 5.5.

5.2 Summarization vs Conclusion Generation

Surprisingly, the summaries performed much better than conclusions, both when used on their own (4-10% better) and very slightly better in combination with the argument (by less than 1%). The former is probably explained by the low conclusion quality that we observed, while the latter could be explained by the fact that in combination the conclusion received much lower weight than the argument itself, masking low conclusion quality.

5.3 Replication on Other Corpora

Both AFS and BWS corpora show marginally worse results across metric variants (by 1% and 2%, respectively) compared to our replication on UKP, showing that the method is generally transferable.

The general patterns with respect to the weighting scheme that we observed on UKP held across the corpora: the concept was the best-performing weighting scheme (average 62%), followed by standard (61%), and then structure (58%). This is also in line with the results reported by Opitz et al. (2021). As for the use of arguments and conclusions / summaries, unlike what we observed on UKP, the argument with summary or conclusion or argument only were all quite close (63% on AFS and 62% on UKP), which is more similar to what Opitz et al. (2021) report than to our replication. Summaries performed slightly worse (58% and 60%, respectively), and the conclusion-only showed the lowest scores (54% and 53%).

Averaging across the corpora, the best metric was using both the argument and the summary (64%), the second best was using the argument with conclusion or argument only (63%), followed by summary only (60%), and conclusion only showed the worst results for us (52%).

5.3.1 Correlation Analysis

Table 3 shows the results of the correlation analysis on the two corpora. Note that the scores reported here are not percentages, but the effect size (correlation strength) multiplied by 100. All the correlations were highly significant, yet the strength of the correlation differed greatly. The patterns observed here match the ones observed for prediction scores, namely that, firstly, summaries performed better than conclusions both when used on their own (by 18 points) and very slightly better in combination with the argument (less than 1 point). Secondly, the concept is the most strongly correlated weighting scheme (averaging 32), followed by standard (28), and then structure (23). Finally, the best metric was using both argument and conclusion / summary or using argument only (36), while using summary only or conclusion only showed worse results (25 and 7, respectively). Comparing the two corpora correlation analysis, one can see that BWS corpus had a little stronger correlations on average (by 4 points).

Notably, the difference between summaries and conclusions is much more pronounced in the correlation analysis as compared to binary prediction scores.

Table 3 additionally shows the correlation analysis results reported by Behrendt and Harmeling (2021) on the two corpora they experimented with (UKP and AFS). As we decided not to run the correlative analysis with discrete UKP scores, only the scores on AFS can be compared. The best-performing model from Behrendt and Harmeling (2021) showed results similar to the best among the ones we obtained (38% vs 39%), theirs being only slightly worse.

5.4 Analysis of Argument Length Effects

Appendix C contains the results of the binary predictions separated by argument length. To carry out the analysis, the sum of the lengths of the two arguments was calculated, and the resulting lengths were binned into separate categories: <100, 100-200, 200-250, 250-300, 300-400, 400-500, and >500. The thresholds obtained at binary prediction evaluation were used to evaluate performance at each length separately. We only analyzed the bins with 50 or more examples. The most populated bins were 250-300 for UKP, 100-200 for AFS, and 300-400 for BWS (see appendix C). To our surprise, the best performance was observed on the length of 100-200 across the corpora. The other patterns were different for the three corpora: for UKP, 100-200 showed 63%, and in the range between 200 and 400 the length had a negative effect in quite a large range for the tree bins (56-61%). For AFS, 100-200 bin performed at 59%, and in the range between 200 and 400 length had a negative effect but all the results are quite close (ranging 55-58%). 400-500 was the third best bin (57%), but also underpopulated (148 examples vs over 1000 in other bins). For BWS 100-200 showed 63% and the length had a negative effect across all bins (ranging 56-63%).

5.5 Conclusion and Summary Analysis

As already mentioned, we obtained lower-quality conclusions and, consequently, our results for conclusions for the argument similarity assessment were also quite low. Therefore, we decided to qualitatively analyze the conclusions and the summaries, as well as compare our conclusions to the ones that we received from the authors of the original paper. We suspect that due to a lack of documentation, our conclusion generation models differed significantly from the one described in the original paper.

We wanted to compare the arguments, the generated summaries and conclusions, as well as the conclusions provided by the authors, and assess the conclusion quality. Tables in the appendix D present these data for all three corpora. Unfortunately, the conclusions for UKP corpus provided to us by the authors were incomplete (did not cover all the arguments) and were also out of order. We attempted to find the matching conclusions.

5.5.1 Summaries

When it came to applying the t5 summarization model, one could see that on the short arguments that we had it worked mostly like a paraphrasing model:

> Argument: "*Abortion is not a question of morality it is a question of providing options to prevent and mitigate risks in certain circumstances.*"

	AFS	BWS	UKP
standard	34,61	37,42	
concept	38,73	43,78	
structure	28,11	30,50	
conclusion_standard	3,95	8,24	
conclusion_concept	6,06	9,35	
conclusion_structure	3,67	7,78	
summary_standard	21,50	29,32	
summary_concept	24,80	33,89	
summary_structure	17,05	22,80	
conclusion_standard_mixed	34,91	37,85	
conclusion_concept_mixed	38,92	**44,08**	
conclusion_structure_mixed	28,49	31,01	
summary_standard_mixed	34,87	37,98	
summary_concept_mixed	**39,01**	44,02	
summary_structure_mixed	28,38	31,08	
argueBERT: average word2vec	11,25		22,29
argueBERT: SBERT	38,02		32,04
argueBERT: sim pred	36,57		**35,33**
argueBERT: order val	38,25		28,41
argueBERT: edge val	36,89		28,36

Table 3: Spearman correlation score multiplied by 100 for the three corpora along with the results reported by Behrendt and Harmeling (2021). The column names correspond to weighting schemes of the AMR argument similarity metric: raw ones standard, concept, and structure are the three weighting schemes applied over argument only, while the ones starting with *conclusion* or *summary* are the weighting schemes for the AMR argument similarity applied over conclusion and summary respectively. The *mixed* feature both the argument and the conclusion or summary, combined with the relative weight of 0.95 given to the argument. We did not calculate the correlation on UKP as we judged 3 discrete values to be insufficient for correlation analysis. The maximal values in each column are highlighted.

> Summary: "*a woman’s abortion is not based on morality but on providing options to mitigate risks.*"

Such summaries were mostly grammatical with some exceptions, especially at the ends of the generated summaries:

> Summary: "*Clones will still be individuals. Clone-clones are still individuals and will continue to be clo*"

> Summary: "*capital punishment is a deterrent greater than life imprisonment. the american system is at best ’fe*".

Notably, there were many cases where the model added a name of a person in front of the summary as if the was this person’s opinion, despite the fact that the person was not mentioned. The people that the model attributed the opinions to the same people over and over:

> Argument: "*In every state that retains the death penalty, jurors have the option of sentencing convicted capital murderers to life in prison without the possibility of parole.*"

> Summary: "*bob greene: jurors can sentence capital murderers to life without parole. he says in every*"

The conclusions provided by the authors of the original paper were actually often very close to the summaries we obtained:

> Argument: "*MSUs power flow controller can be installed anywhere in the existing grid to optimize energy transmission and help reduce transmission congestion.*"

> Summary: "*MSUs power flow controller can be installed anywhere in the existing grid.*"

> Conclusion from Opitz et al. (2021): "*a MSUs power flow controller can be installed anywhere in the existing grid*"

Note that the conclusion by Opitz et al. (2021) was almost identical to our summary. However, there were cases where one or both of the models distorted the argument:

> Argument: "*This could greatly increase the quality of life of those currently living in less than ideal conditions.*"
>
> Summary: "*a recent study found that people living in less than ideal conditions are more likely to die.*"
>
> Conclusion from Opitz et al. (2021): "*a living in less than ideal conditions could greatly improve the quality of life*"

5.5.2 Conclusions

Our conclusion generation model differed greatly from both the summarization and the conclusions from the original article. The conclusions generated by our model were mostly ungrammatical and irrelevant, resembling concatenated fragments of the training or disregarding the argument entirely:

> Argument: "*if a person kills another person it is murder, whether it is sanctioned by the state or not.*"
>
> Conclusion: "*killing someone is a waste of time.When someone kills someone and end his life by dead without killing anyone*"

Here, the model stays at least somewhat close to the topic, but there were examples of absolutely irrelevant conclusions:

> Argument: "*But why do you have to get rid of them by killing them.*"
>
> Conclusion: "*If you do not have a good plan for making your own conclusion, it will be more effective for you to*"

Sometimes a short argument is summarized and concluded into a longer text, for example:

> Argument: "*Clones will still be individuals*"
>
> Summary: "*Clones will still be individuals. Clone-clones are still individuals and will continue to be clo*"
>
> Conclusion: "*Cloning the Cloaks, kneading has become so easy that people can not*"

More examples for each corpus can be found in the appendix D.

The possible causes for this discrepancy in conclusion generation model performance are covered in the limitations section 7.

6 Discussion

The results above indicate that we were, overall, able to reproduce Opitz et al. (2021) and that their method performed comparably on both AFS and BWS corpora. However, we were unable to fully replicate the conclusion generation results due to the lack of documentation for this part of the original paper. Our initial motivation for comparing the performance of the T5 model before and after fine-tuning was to see how much the summaries would turn into conclusions, thus improving the performance. As we explored the summaries compared to the generated conclusions, we observed a much closer performance of summaries to the one reported by the original research. The summaries also closely resembled the conclusions that were generated for the original paper, which we manually inspected and compared. Thus, we can cautiously conclude that the problem lay in the deviations from the undocumented fine-tuning process.

As for the performance on the other corpora and the correlative analysis, we observed slightly lower performance on both corpora. However, the relative patterns observed on binary predictions were confirmed by the correlative analysis: the concept was the best-performing weighing scheme, and using the argument or the argument combined with the summary was the best prediction method. The difference between the summaries and the conclusions was much more pronounced in the correlation analysis than in the binary prediction evaluation. Notably, the AMR argument similarity metric suggested by Opitz et al. (2021) performed slightly better than ArgueBERT on AFS corpus (Behrendt and Harmeling, 2021).

The length analysis revealed unexpected results across the corpora: it was easier for the model to predict the similarity of shorter arguments. It is not quite clear which of the model components were sensitive to length. It is possible that the AMR parser model performs better on shorter sentences similar to the ones it was trained on, or that the S2MATCH metric is better able to capture the similarity of smaller graphs. Unfortunately, as the conclusion generation performed quite poorly overall, we were unable to assess what was the effect of argument length on the contribution of conclusions to the task performance.

It is important to note that the best performance of the AMR-based metric was moderate (67.84% on UKP, 65.34% in AFS, and 63.77% on BWS). It was comparable both to other models (ArgueBERT, Behrendt and Harmeling (2021)) and to simpler metrics like InferSent-fasttext (66.21%) and non-fine-tuned BERT (65.39%) reported by (Opitz et al., 2021) for UKP. This performance on this type of task could be limited by the inter-rater agreement. The upper limit of 78.34% binary prediction performance of human annotators was reported by Daxenberger et al. (2019) for UKP. The correlation-based agreement scores for AFS and BWS were 0.68 Pearson's for AFS and 0.66 Spearman's for BWS (Misra et al., 2016; Thakur et al., 2020a).

7 Limitations and Further Research

The biggest limitations of the present study have to do with the conclusion generation model. As the original paper lacked the information and code for this model, the replication could not be exact. Moreover, as stated above, the task of conclusion generation is an open research question. There can be plenty of valid conclusions derived from a single argument, which makes the majority of models very sensitive to minor changes in the training data. A model with even slightly different training data or variation can produce a vastly different result. Taking into consideration that both the model checkpoints and the code for fine-tuning and inference for the original paper were unavailable, the difference between the original generated conclusions provided by the authors and the ones we obtained is unsurprising. The best example of how the differences in code influence the generations is the `generate` function, part of the T5 model, used for inference/generation. This function is highly dependent on the parameters passed to it. While we kept the parameters that the authors shared with us, it is quite likely that we have a different model and it is thus possible that doing a parameter search to find new parameters that better fit our version would drastically improve the performance of our model.

Furthermore, we deviated from the original study in how the fine-tuning data was generated: while the authors probably excluded premise-premise pairs, we opted to include them in the training data. Very subtle changes like the decisions to separate the premises with a '.' or not, or the position of a white space had a big effect on the model performance. Any of our decisions that differed from the ones made in the original research regarding the fine-tuning dataset construction could have impacted the generated conclusions.

Lastly, there was the issue of evaluating generations. Each step of evaluation needs human overview of the conclusions to determine their quality, as there is no objective metric that could determine the quality automatically. This made the iteration cycles lengthier and more demanding, especially in a low computational resources situation like ours, as we could not explore the effect of batch size. It is quite possible that the model under- or over-fitted to the fine-tuning data with the imperfect evaluation metric.

Overall, the results concerning conclusion generation require further investigation and replication, and should only be regarded as preliminary.

7.1 Further Research

In the wider field of fine-tuned summarization models, specifically, the ones that could be well-suited for very short texts, there are summarization models fine-tuned for paraphrasing. For example, there are transformers of the same type, T5, fine-tuned on the PAWS dataset (Baldridge et al., 2019). The summaries that we obtained for our corpora were effectively paraphrases, so it would be interesting to check if models specifically trained for paraphrasing would perform better. If the conclusion quality would improve, it might also improve overall performance on the argument similarity task.

It is also important to note that AMR parsing might not in itself be the best way to enhance the original sentence information for argument similarity estimations. Although the method is transferable and works comparably on the novel corpora we tried, it still performs fairly low. It would be interesting to compare a wider range of linguistic analyses similar to feature-based approaches or other models for scoring for argument similarity such as BERT-based models similar to ArgueBERT (Behrendt and Harmeling, 2021).

8 Conclusion

This paper presents the results of reproduction, validation, and extension of the AMR argument similarity metric suggested in Opitz et al. (2021). We were able to closely reproduce the results of the paper, and validate the method on two novel corpora, AFS and BWS. The AMR-based metric showed state-of-the-art performance on AFS corpus as compared to ArgueBERT (Behrendt and Harmeling, 2021). The correlation analysis also supported the general patterns observed in Opitz et al. (2021). The biggest limitation of this project was the lack of documentation for conclusion generation in the original paper, which severely limited the model performance for the conclusion-related metrics, while the summaries showed better performance. The effects of length on the argument similarity assessment as well as alternative approaches to the task require further investigation.

Acknowledgments

This project would not have been possible without the helpful suggestions and explanations provided by two of the authors of the original paper, Juri Opitz and Philipp Heinisch.

References

Yamen Ajjour, Henning Wachsmuth, Johannes Kiesel, Martin Potthast, Matthias Hagen, and Benno Stein. 2019. Data acquisition for argument search: The args. me corpus. In *KI 2019: Advances in Artificial Intelligence: 42nd German Conference on AI, Kassel, Germany, September 23–26, 2019, Proceedings 42*, pages 48–59. Springer.

Xiaoyu Bai and Manfred Stede. 2022. Argument similarity assessment in German for intelligent tutoring: Crowdsourced dataset and first experiments. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2177–2187, Marseille, France. European Language Resources Association.

Jason Baldridge, Luheng He, and Yuan Zhang. 2019. Paws: Paraphrase adversaries from word scrambling. In *NAACL*.

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract meaning representation for sembanking. In *LAW@ACL*.

Maike Behrendt and Stefan Harmeling. 2021. Arguebert: How to improve bert embeddings for measuring the similarity of arguments. In *Proceedings of the 17th Conference on Natural Language Processing (KONVENS 2021)*, pages 28–36.

Ralph Bergmann, Mirko Lenz, Stefan Ollinger, and Maximilian Pfister. 2019. Similarity measures for case-based retrieval of natural language argument graphs in argumentation machines. In *FLAIRS Conference*, pages 329–334.

Deng Cai and Wai Lam. 2020. Amr parsing via graph-sequence iterative inference.

Shu Cai and Kevin Knight. 2013. Smatch: an evaluation metric for semantic feature structures. In *Annual Meeting of the Association for Computational Linguistics*.

Carlos Iván Chesnevar and Ana G Maguitman. 2004. Arguenet: An argument-based recommender system for solving web search queries. In *2004 2nd International IEEE Conference on'Intelligent Systems'. Proceedings (IEEE Cat. No. 04EX791)*, volume 1, pages 282–287. IEEE.

Johannes Daxenberger, Steffen Eger, and Iryna Gurevych. 2019. Ukp aspect corpus.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Bill Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Third International Workshop on Paraphrasing (IWP2005)*.

Rebecca Dridan and Stephan Oepen. 2011. Parser evaluation using elementary dependency matching. In *Proceedings of the 12th International Conference on Parsing Technologies*, pages 225–230, Dublin, Ireland. Association for Computational Linguistics.

Jeffrey Flanigan, Sam Thomson, Jaime G Carbonell, Chris Dyer, and Noah A Smith. 2014. A discriminative graph-based parser for the abstract meaning representation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1426–1436.

Shai Gretz, Yonatan Bilu, Edo Cohen-Karlik, and Noam Slonim. 2020. The workweek is the best time to start a family – a study of GPT-2 based claim generation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 528–544, Online. Association for Computational Linguistics.

Timon Gurcke, Milad Alshomary, and Henning Wachsmuth. 2021. Assessing the sufficiency of arguments through conclusion generation. In *Proceedings of the 8th Workshop on Argument Mining*, page 67–77.

Philipp Heinisch, Anette Frank, Juri Opitz, and Philipp Cimiano. 2022. Strategies for framing argumentative conclusion generation. In *Proceedings of the 15th International Conference on Natural Language Generation*, pages 246–259, Waterville, Maine, USA and virtual meeting. Association for Computational Linguistics.

Irene Langkilde and Kevin Knight. 1998. Generation that exploits corpus-based statistical knowledge. In *COLING 1998 Volume 1: The 17th International Conference on Computational Linguistics*.

Mirko Lenz, Stefan Ollinger, Premtim Sahitaj, and Ralph Bergmann. 2019. Semantic textual similarity measures for case-based retrieval of argument graphs. In *Case-Based Reasoning Research and Development: 27th International Conference, ICCBR 2019, Otzenhausen, Germany, September 8–12, 2019, Proceedings 27*, pages 219–234. Springer.

Amita Misra, Pranav Anand, Jean E Fox Tree, and Marilyn Walker. 2017. Using summarization to discover argument facets in online ideological dialog. *arXiv preprint arXiv:1709.00662*.

Amita Misra, Brian Ecker, and Marilyn Walker. 2016. Measuring the similarity of sentential arguments in dialogue. pages 276–287.

Juri Opitz, Anette Frank, and Letitia Parcalabescu. 2020. Amr similarity metrics from principles. *Transactions of the Association for Computational Linguistics*, 8(0).

Juri Opitz, Philipp Heinisch, Philipp Wiesenbach, Philipp Cimiano, and Anette Frank. 2021. Explainable unsupervised argument similarity rating with Abstract Meaning Representation and conclusion generation. In *Proceedings of the 8th Workshop on Argument Mining*, pages 24–35, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. pages 1–53.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

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.

Frane Šarić, Goran Glavaš, Mladen Karan, Jan Šnajder, and Bojana Dalbelo Bašić. 2012. TakeLab: Systems for measuring semantic text similarity. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 441–448, Montréal, Canada. Association for Computational Linguistics.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31.

Christian Stab and Iryna Gurevych. 2017. Parsing argumentation structures in persuasive essays. *Computational Linguistics*, 43(3):619–659.

Shahbaz Syed, Khalid Al Khatib, Milad Alshomary, Henning Wachsmuth, and Martin Potthast. 2021. Generating informative conclusions for argumentative texts. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3482–3493, Online. Association for Computational Linguistics.

Nandan Thakur, Johannes Daxenberger, and Iryna Gurevych. 2020a. Bws argument similarity corpus.

Nandan Thakur, Nils Reimers, Johannes Daxenberger, and Iryna Gurevych. 2020b. Augmented sbert: Data augmentation method for improving bi-encoders for pairwise sentence scoring tasks. *arXiv preprint arXiv:2010.08240*.

Chris van der Lee, Albert Gatt, Emiel van Miltenburg, Sander Wubben, and Emiel Krahmer. 2019. Best practices for the human evaluation of automatically generated text. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 355–368, Tokyo, Japan. Association for Computational Linguistics.

Henning Wachsmuth, Shahbaz Syed, and Benno Stein. 2018. Retrieval of the best counterargument without prior topic knowledge. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 241–251, Melbourne, Australia. Association for Computational Linguistics.

Wenhao Yu, Chenguang Zhu, Zaitang Li, Zhiting Hu, Qingyun Wang, Heng Ji, and Meng Jiang. 2022. A survey of knowledge-enhanced text generation. *ACM Comput. Surv.*, 54(11s).

A Fine-Tuning

The information on the conclusion generation model provided by the authors was:

- the model: "`t5-small`"
- the size: "all conclusions (for fine-tuning) were truncated after 128 `T5-small-tokenizer-wordpieces`"
- the training parameters: "`AdamW( learning_rate = ExponentialDecay (initial_learning_rate=3e-4, decay_rate=0,95, decay_steps=100, staircase=True), weight_decay = ExponentialDecay (initial_learning_rate=3e-7, decay_rate=0,95, decay_steps=80, staircase=True))`"
- the training: "epoch-based early stopping mechanism, stopping the fine-tuning w.r.t a minimal cross-entropy-loss on the validation split".
- the inference: "`generate( samples_enc["input_ids"], max_length=25, min_length=4, do_sample=True, num_beams=4, top_k=20, temperature=.75, no_repeat_ngram_size=2, repetition_penalty=1.25) ,skip_special_tokens=True)`"

B Detailed Results

	original-UKP	**UKP**	**AFS**	**BWS**
standard	65,44	66,09	63,48	61,75
concept	68,17	**67,84**	**65,34**	62,96
structure	60,74	61,99	60,50	60,11
conclusion_standard	57,31	51,18	53,72	51,95
conclusion_concept	60,29	50,58	54,01	53,02
conclusion_structure	56,48	50,62	53,44	52,53
summary_standard		61,51	58,17	59,91
summary_concept		63,90	59,59	61,71
summary_structure		57,42	55,83	57,41
conclusion_standard_mixed	66,21	65,58	63,81	61,22
conclusion_concept_mixed	**68,70**	67,18	64,57	**63,77**
conclusion_structure_mixed	61,14	62,30	60,92	60,13
summary_standard_mixed		66,82	63,60	62,41
summary_concept_mixed		67,40	64,91	63,15
summary_structure_mixed		62,58	60,52	59,62
random	48,01			
tf-idf	61,18			
InfSnt-fText	66,21			
InfSnt-GloVe	64,94			
GloVe Emb	64,68			
ELMo Emb	64,47			
BERT Emb	65,39			
Human	***78,34***			

Table 4: f1 score on the three corpora along with the results reported by Opitz et al. (2021). The column names correspond to weighting schemes of the AMR argument similarity metric: raw ones standard, concept, and structure are the three weighting schemes applied over argument only, while the ones starting with *conclusion* or *summary* are the weighting schemes for the AMR argument similarity applied over conclusion and summary respectively. The *mixed* feature both the argument and the conclusion or summary, combined with the relative weight 0.95 given to the argument. The maximal values in each column are highlighted.

C Analysis by Length

Table 5: **UKP corpus.**
Analysis of the influence of length on AMR argument similarity metric performance on UKP corpus. The maximal values in each row are highlighted (the values in the first and the last two columns are disregarded).

	<100	**100-200**	**200-250**	**250-300**	**300-400**	**400-500**	>500
standard	1,00	**0,70**	**0,70**	0,66	0,65	0,83	
concept	1,00	**0,73**	0,71	0,69	0,65	1,00	
structure	1,00	0,63	**0,67**	0,62	0,60	**0,67**	
conclusion_standard	0,00	**0,61**	0,50	0,51	0,51	0,33	
conclusion_concept	0,00	**0,61**	0,52	0,51	0,49	0,33	
conclusion_structure	0,00	**0,57**	0,49	0,51	0,51	0,40	
summary_standard	1,00	**0,70**	0,67	0,61	0,59	0,78	
summary_concept	1,00	**0,72**	0,68	0,63	0,61	0,78	
summary_structure	1,00	**0,63**	0,60	0,58	0,55	0,83	
conclusion_standard_mixed	1,00	**0,71**	0,70	0,65	0,64	0,83	
conclusion_concept_mixed	1,00	**0,46**	**0,46**	0,43	0,42	0,40	
conclusion_structure_mixed	1,00	0,60	**0,64**	0,59	0,59	0,49	
summary_standard_mixed	1,00	**0,71**	**0,71**	0,66	0,65	0,83	
summary_concept_mixed	1,00	0,46	**0,50**	0,45	0,43	0,40	
summary_structure_mixed	1,00	0,59	**0,63**	0,57	0,57	0,49	
mean	0,80	**0,63**	0,61	0,58	0,56	0,62	
count	1	247	847	**1473**	1021	6	0

Table 6: **AFS corpus.**
Analysis of the influence of length on AMR argument similarity metric performance on AFS corpus. The maximal values in each row are highlighted (disregarding the values for the length above 500).

	<100	**100-200**	**200-250**	**250-300**	**300-400**	**400-500**	**>500**
standard	0,44	**0,67**	0,66	0,64	0,60	0,63	0,40
concept	0,56	0,68	0,67	0,65	0,60	**0,69**	0,40
structure	0,50	**0,64**	**0,64**	0,61	0,58	0,58	0,40
conclusion_standard	0,14	0,55	0,55	0,51	0,55	**0,56**	0,40
conclusion_concept	0,14	**0,55**	**0,55**	0,51	**0,55**	**0,55**	0,40
conclusion_structure	0,14	0,55	0,54	0,51	0,55	**0,56**	0,40
summary_standard	0,62	**0,63**	0,61	0,55	0,54	0,50	0,40
summary_concept	**0,70**	0,65	0,62	0,57	0,55	0,51	1,00
summary_structure	**0,62**	0,59	0,58	0,56	0,52	0,46	0,40
conclusion_standard_mixed	0,44	0,67	0,65	0,64	0,60	**0,69**	0,40
conclusion_concept_mixed	0,45	0,45	0,46	0,47	0,47	**0,52**	0,40
conclusion_structure_mixed	0,31	0,54	0,56	0,54	0,52	**0,58**	1,00
summary_standard_mixed	0,50	**0,68**	0,66	0,64	0,60	0,61	0,40
summary_concept_mixed	0,48	0,45	0,46	0,47	0,47	**0,52**	0,40
summary_structure_mixed	0,38	0,54	0,55	0,53	0,50	**0,59**	1,00
mean	0,43	**0,59**	0,58	0,56	0,55	0,57	0,52
count	12	**1625**	1457	1403	1352	148	3

Table 7: **BWS corpus.**
Analysis of the influence of length on AMR argument similarity metric performance on BWS corpus. The maximal values in each row are highlighted.

	<100	**100-200**	**200-250**	**250-300**	**300-400**	**400-500**	**>500**
standard	0,60	**0,65**	0,63	0,59	0,62	0,59	0,60
concept	0,66	**0,67**	0,63	0,62	0,64	0,62	0,61
structure	**0,66**	0,62	0,59	0,60	0,60	0,60	0,56
conclusion_standard	0,52	0,51	0,53	0,51	0,50	**0,57**	0,49
conclusion_concept	0,43	0,51	0,54	0,51	0,53	**0,55**	**0,55**
conclusion_structure	0,58	0,53	0,53	0,55	0,52	**0,58**	0,44
summary_standard	0,60	**0,65**	0,62	0,58	0,59	0,58	0,51
summary_concept	**0,70**	0,68	0,64	0,59	0,61	0,57	0,54
summary_structure	**0,66**	0,60	0,58	0,55	0,57	0,57	0,50
conclusion_standard_mixed	0,58	**0,65**	0,63	0,59	0,61	0,60	0,61
conclusion_concept_mixed	0,56	**0,60**	0,57	0,51	0,49	0,41	0,32
conclusion_structure_mixed	0,60	0,62	0,59	0,59	0,60	0,60	**0,63**
summary_standard_mixed	0,58	**0,64**	0,63	0,59	0,62	0,60	0,62
summary_concept_mixed	0,58	0,58	0,53	0,48	0,45	0,36	0,25
summary_structure_mixed	0,60	0,61	0,59	0,58	0,59	0,60	**0,63**
mean	0,59	**0,61**	0,59	0,56	0,57	0,56	0,52
count	39	597	605	608	**845**	443	263

D Conclusion Analysis

Table 8: **UKP Corpus.**
The arguments, generated summaries, generated conclusions, and the conclusions generated for the original paper (Opitz et al., 2021). The pair and the sentence number indicate the id of the example in the dataset (starting at 0).

pair	**sent**	**argument**	**summary**	**conclusion**	**conclusions by Opitz et al. (2021)**
1	1	This could greatly increase the quality of life of those currently living in less than ideal conditions.	a recent study found that people living in less than ideal conditions are more likely to die.	Having professional training help people increase their productivity	a living in less than ideal conditions could greatly improve the quality of life
3594	1	And every wind turbine slows the wind, thus reducing the wind energy available to any downwind turbines.	every wind turbine slows the wind, thus reducing wind energy available to any downwind turbines.	Wind farms, which is a source of energy, can depress more than any other household item	wind turbines are a great way to reduce the wind energy
3592	1	MSUs power flow controller can be installed anywhere in the existing grid to optimize energy transmission and help reduce transmission congestion.	MSUs power flow controller can be installed anywhere in the existing grid.	Many people, for example, are suffering from allergy because of these gasses	a MSUs power flow controller can be installed anywhere in the existing grid
503	2	[Then], as the currency matures, we can extend, so people can pay their rents and utilities, [or] pay for food through this sort of service.	bob greene: as currency matures, we can extend so people can pay rents and utilities. green	If people in rich countries ponder poor people suffering from poverty and diseases, they would not have to go back	the currency is a good way for people to pay their rents
2211	1	I do not have a sense of how much damage it will do.	I do not have a sense of how much damage it will do, says cnn's	I was shocked by the cruelty which I had never done before	I do not have a sense of how much damage it will do
38	2	They hope that one day, these robots can be used to detect and/or neutralize specific chemical and toxins found in our environment.	robots hope to be used to detect and/or neutralize specific chemical and toxins found in our environment.	This is due to the limitation of work overload	robots can be used to detect and/or neutralize specific chemical

Table 9: **AFS Corpus.**
The arguments, generated summaries, and generated conclusions. The pair and the sentence number indicate the id of the example in the dataset (starting at 0). The three topics are combined into one dataset.

pair	sent	argument	summary	conclusion
0	1	Yes there should be a death penalty but it should only be used in extreme circumstances like treason, mass murder, ordering murders from within prison, and killing someone in prison.	lz granderson: death penalty should only be used in extreme circumstances. he says it's not	it is an oppertunity to find a way to be murdered.by doing something which is
1	1	In every state that retains the death penalty, jurors have the option of sentencing convicted capital murderers to life in prison without the possibility of parole.	bob greene: jurors can sentence capital murderers to life without parole. he says in every	if a criminal is convicted, he will have the chance to be found guilty compared to
2	1	so you weigh whatever monetary savings (if any) over the life of someone found innocent after conviction (even if rare)?	bob greene: you weigh what monetary savings (if any) over the life of someone found innocent.	If someone faced so complicated confusion in his study and his many endeavors on making out it were miscarried,
5997	1	In America, every state that has enacted restrictive gun control has seen an increase in crime, police brutality, questionable law enforcement practices such as violations of 4th amendment rights and so on.	bob greene: every state that has enacted restrictive gun control has seen an increase in crime. green	fewer guns available mean less crime
793	1	But why do you have to get rid of them by killing them.	Why do you have to get rid of them?,	If you do not have a good plan for making your own conclusion, it will be more effective for you to
1484	1	killing a person, no matter the reason, is still murder.	killing a person, no matter the reason, is still murder. daniel treisman:	killing someone is a waste of time.As he said, killing everyone is dead.If we kill,
1484	2	if a person kills another person it is murder, whether it is sanctioned by the state or not.	if a person kills another person it is murder, whether it's sanctioned by the state or	killing someone is a waste of time.When someone kills someone and end his life by dead without killing anyone
495	2	the use of the death penalty is an act of revenge which cannot be undone, so should the state be put on trial for putting to death an innocent person?	sally kohn: death penalty is an act of revenge which cannot be undone. she says it	the death penalty is a form of revenge which encourages the criminal to be convicted.it can be replaced

Table 10: **BWS Corpus.**
The arguments, generated summaries, and generated conclusions. The pair and the sentence number indicate the id of the example in the dataset (starting at 0).

pair	sent	argument	summary	conclusion
1	1	Abortion is not a question of morality it is a question of providing options to prevent and mitigate risks in certain circumstances.	a woman's abortion is not based on morality but on providing options to mitigate risks.	Having professional training help employees gain more confidence in their business
2	1	They also add that if abortion was illegal , the procedure would be performed , regardless – the only difference being that it would be performed under dangerous , substandard conditions	abortion would be performed if it was illegal. only difference is that it would take place under dangerous, sub	abortion is a form of revenge which creates the sexual relationship with the child
3397	1	Proponents have found a significant positive impact on school climate , safety , and students ' self-perception from the implementation of uniforms.	uniforms have a significant positive impact on school climate, safety and students.	Co-operation is essential for teamwork.co-operative approach will save time and considerable expense.long-d
119	2	to prohibit the application of genetic engineering techniques that may be contrary to human dignity	human dignity is violated by cloning, says christopher stanley.	Similarly to the ivory-towers, many people are also human beings.We should put our own
979	2	Strict gun control laws do not work in Mexico, and will not work in the United States.	strict gun control laws do not work in Mexico, and won't work.	there are some reason why the level of bloody increases significantly in the communities
85	1	It is hard to say whether this is true , but we have seen devastating consequences of other research-programs , even with good intentions , such as nuclear research.	we have seen devastating consequences of other research-programs.	If you want to get brilliant achievements such as going to college, acing your school's final tests
85	2	Clones will still be individuals	Clones will still be individuals. Clone-clones are still individuals and will continue to be clo	Cloning the Cloaks, kneading has become so easy that people can not
1181	2	"If capital punishment can be a deterrent greater than life imprisonment at all , the American system is at best a feeble one."	capital punishment is a deterrent greater than life imprisonment. the american system is at best 'fe	: Capital punishment helps to safeguard human rights in a strict and effective manner.capital punishment protects the society from