Full-Text Argumentation Mining on Scientific Publications

Arne Binder[1] **Bhuvanesh Verma**[2] **Leonhard Hennig**[1]
[1]German Research Center for Artificial Intelligence (DFKI)
[2]University of Potsdam
[1]{*arne.binder, leonhard.hennig*}*@dfki.de*
[2]*bhuvanesh.verma@uni-potsdam.de*

Abstract

Scholarly Argumentation Mining (SAM) has recently gained attention due to its potential to help scholars with the rapid growth of published scientific literature. It comprises two subtasks: argumentative discourse unit recognition (ADUR) and argumentative relation extraction (ARE), both of which are challenging since they require e.g. the integration of domain knowledge, the detection of implicit statements, and the disambiguation of argument structure (Al Khatib et al., 2021). While previous work focused on dataset construction and baseline methods for specific document sections, such as abstract or results, full-text scholarly argumentation mining has seen little progress. In this work, we introduce a sequential pipeline model combining ADUR and ARE for full-text SAM, and provide a first analysis of the performance of pretrained language models (PLMs) on both subtasks. We establish a new SotA for ADUR on the Sci-Arg corpus, outperforming the previous best reported result by a large margin (+7% F1). We also present the first results for ARE, and thus for the full AM pipeline, on this benchmark dataset. Our detailed error analysis reveals that non-contiguous ADUs as well as the interpretation of discourse connectors pose major challenges and that data annotation needs to be more consistent.

1 Introduction

Argumentation Mining (AM) is concerned with the detection of the argumentative structure of text (Stede and Schneider, 2018). It is commonly organized into two subtasks: 1) Recognition of argumentative discourse units (ADUs), i.e. detecting argumentative spans of text and classifying them into types such as *claim* or *premise*, and 2) determining which ADUs have a relationship to each other and of what kind, e.g. *support* or *attack*. Consider the following example, where the premise P supports the claim C:

> Dot-product attention is much faster than additive attention$_C$, since it can be implemented using highly optimized matrix multiplication code$_P$.[1]

Since the amount of published scientific literature is growing exponentially (Fortunato et al., 2018), there is recently an increased interest in scholarly argumentation mining (SAM). Understanding the argumentative structure is key, not just to efficiently digest such work, but also to assess its quality (Walton, 2001). Solving scholarly AM is challenging, because it requires, among other things, the use of domain knowledge, the detection of implicit statements, and the disambiguation of argument structure (Al Khatib et al., 2021). This is even harder when handling full-text that is often less concise and standardized, than, for example, abstracts.

Previous work in SAM has focused on dataset construction (Teufel and Moens, 1999; Lauscher et al., 2018b), ADU recognition (Lauscher et al., 2018a; Li et al., 2021), and the analysis of specific document sections, such as abstract or results (Dasigi et al., 2017; Accuosto and Saggion, 2019; Mayer et al., 2020). However, to get a thorough understanding of a scientific publication, all parts of the document matter. Ideally, they back up the main argumentation and usually contain details that are relevant for the knowledgeable reader, thus, they should not be neglected. However, since the task is very complex, also for humans, there is not much training data for full-text SAM available.

Pretrained Language Models (PLMs) such as SciBERT (Beltagy et al., 2019) may help to address the above challenges because they contain a lot of linguistic and domain knowledge and have better long-range capabilities, allowing for improved contextualisation, especially when training data is rare. We hence propose a PLM based model for full-text

[1]replicated from Vaswani et al. (2017)

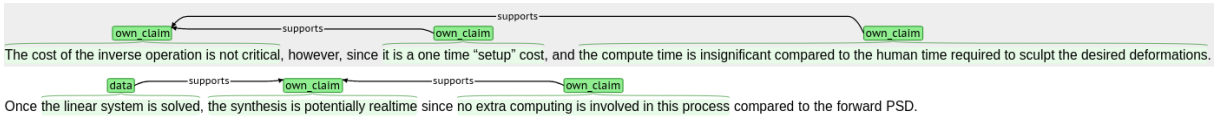


Figure 1: Example with argumentative structure from the Sci-Arg dataset.

SAM. To summarize, our contributions in this work are:

- We are the first to investigate PLMs for full-text SAM, and to present a sequential pipeline for both ADU recognition and argumentative RE on full-text scientific publications (Section 3).
- Our experimental results show that a SciBERT-based ADU recognition model improves over the state-of-the-art by $+7\%$ F1-score. We present the first relation extraction baseline for the Sci-Args corpus and achieve strong 0.74 F1 (Section 5.1).
- Our detailed error analysis reveals open challenges and possible ways of improvements (Section 5.2).

2 Preliminaries

We first define the two tasks of ADUR and ARE, and discuss differences to the standard Information Extraction (IE) tasks of Named Entity Recognition (NER) and Relation Extraction (RE).

An Argumentative Discourse Unit (ADU) can be defined as "span of text that plays a single role for an argument being analyzed and is demarcated by neighboring text spans that play a different role, or none at all" (Stede and Schneider, 2018). It is the smallest unit of argumentation, and may span anything from an in-sentence clause up to multiple full sentences. ADU recognition requires both detecting argumentative spans, as well as classifying them into predefined categories. Typically, this is realised as sequence tagging task similar to NER, where a sequence of tokens $X = \{t_1, t_2, ..., t_N\}$ is assigned with a corresponding N-length sequence of labels $Y = \{l_1, l_2, ..., l_N\}$ with $l_i \in C$ where C is the set of tags that result from converting the ADU types into a tagging scheme like BIO2.[2] In scholarly AM, common ADU classes are *(Own / Background) Claim*, and *Evidence*, *Data*, or *Warrant* (Green, 2014; Lauscher et al., 2018b).

[2]BIO2: **B**egin, **I**nside, **O**utside of an entity

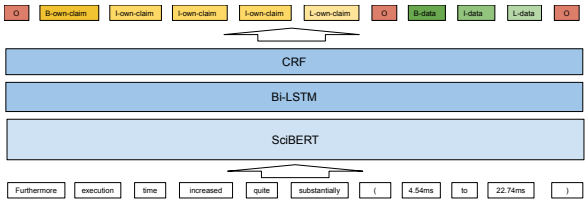
In contrast to NER, ADUs typically vary much more in length than named entities. They are also highly context dependent and often discontinuous. ADUR is also related to discourse segmentation, but depends more on broader context and semantics instead of linguistic structure. Elementary Discourse Units (EDUs), the building blocks in the context of Rethorical Structure Theory (Mann and Thompson, 1988), are more fine-grained, of shorter length and usually cover the complete text which is less the case for argumentative units.

Argumentative Relation Extraction is usually defined as classifying a pair of ADUs, *head* and *tail*, as either an instance of one of the target types or the artificial NO-RELATION type. In other words, the task is to assign a label $Y \in C \cup \{\text{NO-RELATION}\}$ to a given input $X = \{T, h, t\}$, where C is the set of relation types, T is the text and $h = (s_h, e_h, l_h)$ and $t = (s_t, e_t, l_t)$ describe the candidate head and tail entities where s and e are the start and end indices with respect to T and l is the entity type. Typical relation types for SAM are *Supports*, *Mentions*, *Attacks*, *Contradicts*, and *Contrasts* (Lauscher et al., 2018b; Accuosto and Saggion, 2019; Nicholson et al., 2021).

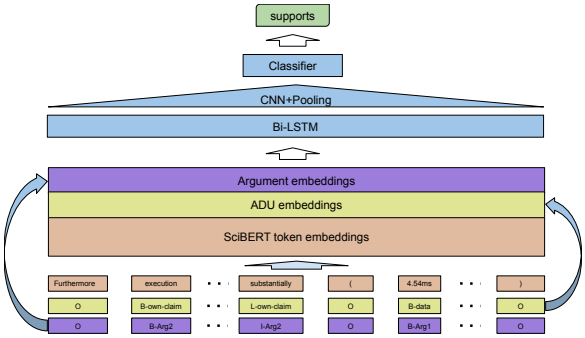
ARE is very similar to standard RE, but SAM relations are often marked by syntactic cues such as connectors, e.g. "because", "however", or "but", whereas in common RE, content words like verbs and nouns are typical relation triggers. This makes ARE challenging because these connectors do not always realise argumentative structure, but also mark other aspects of discourse. Consider, for example, the different meanings of "while" in the following example:

1. While I love a romantic dinner, I also like fast food.
2. While I prepare dinner, I watch a movie.

Here, the "while" in sentence 1) has a contrastive meaning, whereas sentence 2) denotes a temporal aspect.



(a) **ADU Recognition.** Tokens are embedded with a frozen PLM, further contextualized with a trained LSTM followed by a CRF to calculate the tag sequence.



(b) **Argumentative RE.** Tokens are embedded with a frozen PLM, ADU tags and argument tags are embedded with simple embedding matrices. Embeddings are concatenated, contextualized with a LSTM and converted into a single vector that gets classified by a single fully connected layer.

Figure 2: Model setup for (a) ADUR (top) and (b) ARE (bottom).

3 Models

We propose a pipeline of two distinct models, one for each subtask, that are described in the following.

ADU Recognition (ADUR). The architecture of the ADUR model is visualized in Figure 2a. We first embed the token sequence with a frozen PLM encoder. For sequences that exceed the maximum input length of the embedding model, we process the sequence piece-wise and concatenate the result afterwards. The embedded tokens are then fed into a BiLSTM (Schuster and Paliwal, Nov./1997). Finally, a Conditional Random Field (CRF) (Lafferty et al., 2001) is used to obtain the label probabilities for each token. We use a combination of a frozen PLM with a trainable contextualization (LSTM) on top because its training requires less resources than fine-tuning the PLM and initial tests have shown similar performance.[3]

Argumentative RE (ARE). The model architecture for the relation extraction subtask is shown in Figure 2b. ARE is implemented as a classification task, where a pair of candidate ADUs is selected and marked in the input token sequence. To reduce combinatorial complexity, only ADU pairs with a distance smaller than some threshold `d` are considered. Similar to ADU recognition, we first embed the token sequence in a window of `k` tokens around the candidate entity pair with a frozen PLM model. We also create non-contextualized embeddings for the ADU- and argument-tags of the tokens within the window. As argument tags we simply use *head* and *tail* labels to mark the candidate entity tokens. All three embedding sequences are concatenated token-wise and fed into a BiLSTM. The result is converted into a single vector using a Convolutional Neural Network (CNN) and max-pooling, which then is classified as one of the relation labels by a linear projection with softmax.

[3]Note that the training dataset is relative small, so restricting the number of trainable parameters seems to mitigate overfitting.

	Train	Test	Total
ADUs			
background claim	2563	661	3224
own claim	4608	1241	5849
data	3346	858	4204
Relations			
supports	4426	1260	5686
contradicts	551	133	684
semantically same	36	3	39
parts of same	1000	269	1269

Table 1: Label counts for the Sci-Arg dataset.

4 Experimental Setup

Dataset. We use the Sci-Arg dataset (Lauscher et al., 2018b) for model training and evaluation. It is the only available full text argumentation mining dataset for scientific publications. It contains 40 full text publications annotated with ADUs and argumentative relations. Figure 1 shows an example excerpt, and Table 1 summarizes the main dataset statistics. The PARTS OF SAME relation type is used to model non-contiguous spans. The label counts differ slightly from values published in Lauscher et al. (2018b), because annotations in one file (A28) caused parsing errors and were excluded. Furthermore, non-contiguous spans are not merged. We create a train/test split by using the first 30 documents for training and the remaining 9 for evaluation.

system	span based		token based
	exact	weak	
Lauscher 2018c	-	-	0.447
ours	0.532	0.668	0.518
human	0.602	0.729	-

Table 2: **ADU Recognition Performance** as F1 macro average over classes. For *weak* metrics, the gold and the predicted span have to match for at least the half of the characters of the longer span.

	F1-exact	F1-weak
@gold ADUs	0.739	
@predicted ADUs	0.210	0.310
human	0.341	0.469

Table 3: **Argumentative RE Performance** as micro average over classes with provided gold ADUs (@gold ADUS) or ADUs predicted with our entity recognition model (@predicted ADUs), i.e. the full relation extraction pipeline. *human* indicates inter-annotator-agreement for the corpus data (Lauscher et al., 2018c) which is comparable to *@predicted ADUs*. For *weak* metrics, best weakly matching ADUs are calculated first, then predicted relations are mapped to these and finally metrics are calculated as usual.

Preprocessing. We preprocess the documents by removing the initial XML headers. To decrease the sequence length of the input, we also split the documents into sections, e.g. *introduction* or *conclusion*. This is important to lower computational resource consumption since recent PLMs like SciBERT (Beltagy et al., 2019) usually scale quadratic with the input length and are restricted to a certain max input size, e.g. 512 tokens. Unfortunately, this leads to the removal of all relations labeled with SEMANTICALLY SAME, since these connect ADUs from different sections. However, this affects only 0.6% of the argumentative relations instances.

Data Augmentation. If the pair of ADUs (A, B) is part of an argumentative relation, it is wrong to assume that B is argumentatively unrelated with A, i.e. (B, A) should not be in the NO RELATION class. Thus, we add reversed instances for each available relation in the dataset with the special label SUPPORTS REV in the case of SUPPORTS and keep the labels for CONTRADICTS and PARTS OF SAME since these relations are symmetric. In addition to the positive training instances, we also sample negative relation instances from all possible ADU pairs that are no instances of any argumentative relation.

Training Objective. We use the the cross entropy loss (Rubinstein, 1999) as the training objective for both models f_{ADU} and f_{RE}:

$$\mathcal{L}_{CE}(y, \hat{y}; \boldsymbol{\theta}) := -f_{\boldsymbol{\theta}}(y) \cdot \log f_{\boldsymbol{\theta}}(\hat{y})$$

where y and $\hat{y}$ are the target and predicted probabilities for the token or relation labels, respectively, and $\boldsymbol{\theta}$ is the set of trainable model parameters. In the case of ADU recognition, we obtain the best tagging sequence via Viterbi Decoding (Viterbi, 1967), as usual for CRF-based models.

Metrics. Since we compare against evaluation results from Lauscher et al. (2018d), we adopt their metrics for ADU recognition, namely a token-based F1-score that is macro-averaged over classes. However, we also compute *span-based* macro-F1 scores in two variants as described in Lauscher et al. (2018b): For *exact* span-based metrics, the recognized ADU has to match exactly for start and end indices, as well as ADU type. For *weak* matches, the ADU has to match in type, but the target and predicted spans only have to overlap by at least the half of the length of the shorter span. Weak match evaluation is motivated by considerable length and variance of ADU expressions, which makes exact matches difficult, and also allows for comparison with human annotator agreement scores as presented in Lauscher et al. (2018c).

For the relation recognition task, we follow the literature and present micro-averaged F1 scores. Similar to ADU recognition metrics, we calculate weak metrics by first determining target ADUs that can be assigned to predicted ADUs in the way of weak ADU matching as described above, and then calculate F1 scores as usual (Lauscher et al., 2018b). Note that PARTS OF SAME is just a helper relation, so we merge ADUs connected by this relation type first, and then compute scores over the remaining relation types.

Training Details & Hyperparameters. For both tasks, we first conduct a hyperparameter search. We use token-based macro-F1 as the optimization target for ADU recognition and micro-F1 as the target for relation classification. Final hyperparameter values are listed in Appendix A.2.

		P	R	F1
exact	background claim	0.56	0.44	0.49
	own claim	0.48	0.55	0.51
	data	0.57	0.62	0.60
weak	background claim	0.77	0.60	0.68
	own claim	0.63	0.73	0.67
	data	0.62	0.69	0.65

Table 4: **ADUR Performance per Class.** Macro averaged precision (P), recall (R), and F1.

Since there is no dev split, we perform 5-fold cross validation for each subtask on the train split with the best hyperparameter settings and different random seeds for parameter initialization. The best of these 5 models are used for the final evaluation. Detailed training configurations, logs and statistics for the ADU recognition and the ARE subtasks are collected within the Weights & Biases framework.[4] We make these and our source code publicly available for better reproducibility of our experiments.[5]

5 Results and Discussion

This section presents our experimental results. First, we compare against the ADU recognition baseline as provided by Lauscher et al. (2018c). Then, we present findings about prominent error cases and close with an ablation study.

5.1 Results

Table 2 presents the macro-F1 scores of the ADUR baseline, our approach, and human performance in terms of inter-annotator agreement, as reported in Lauscher et al. (2018c). Our model achieves 0.518 token-based F1, significantly outperforming the baseline by 7%. The gap to the human performance is also narrow, especially when looking at the weak metrics with relaxed boundary constraints, where our model achieves 92% of to the human score. For exact metrics, the model reaches only 88% of the human performance, suggesting that exact ADU boundary detection is more challenging. The performance of the model for argumentative RE is a strong 0.739 micro-F1. Note, that we need to merge non-contiguous ADUs first before calculating the ARE scores. We do this via predicted PARTS OF SAME relations, which are recognized with a F1-score of 0.860. For the full pipeline, the model achieves a respectable 0.210 micro-F1 score, which corresponds to 62% of the human performance.

[4] see https://wandb.ai

[5] For ADU recognition, see https://wandb.ai/sam_dfki/best_adu_uncased, for argumentative RE, see https://wandb.ai/sam_dfki/best_rel_uncased, and for the source code, see https://github.com/DFKI-NLP/sam.

	P	R	F1
contradicts	0.505	0.724	0.595
supports	0.739	0.774	0.756

Table 5: **ARE Performance per Class**. Micro averaged precision (P), recall (R), and F1 on gold ADUs. Note that non contiguous ADUs linked via predicted PARTS OF SAME relations are merged first before calculating the scores.

5.2 Error Analysis

ADUR Error Analysis. The decrease in performance when comparing weak with exact metrics is high for the classes BACKGROUND CLAIM (-28%) and OWN CLAIM (-24%), but low for class DATA (-8%), see Figure 4. This may be because the latter is mainly about references or mentions of concise facts where boundaries are much easier to detect.

Most of the errors originate from *detecting* ADUs, i.e. deciding if a text span is an ADU from any type, in comparison to *classifying* a detected ADU span into one of types. The exact span-based macro-F1 for the subtask of ADU *classification* is 0.854, whereas the respective score for ADU *detection* is only 0.617. This difference is even larger for the RE subtask where the micro-F1 is 0.749 for relation *detection* and 0.992 (!) for relation *classification*. Figure 3 shows the confusion matrices for the ADUR and ARE subtasks.

Interestingly, many of ADU classification errors (48%) are instances of type BACKGROUND CLAIM, where the model predicts OWN CLAIM instead, indicated by low precision for OWN CLAIM and low recall for BACKGROUND CLAIM as shown in Table 4. Looking into these misclassifications revealed the following main challenges (in order of decreasing frequency): 1) an island of one or two background claims surrounded by many own claims or located at the border between regions of these two types, 2) the ADU is linked via the structural PARTS OF SAME relation, i.e. it is split by some other content and at least one part of the complete ADU is

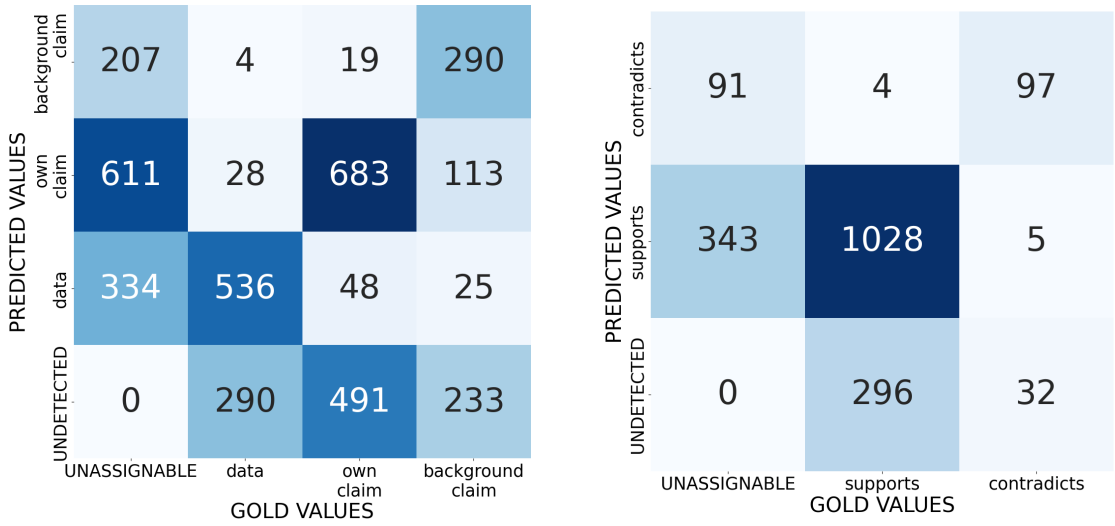


Figure 3: Confusion matrices for ADU recognition (left) and argumentative RE (right).

not detected correctly, and 3) mentions of the author in a background claim (e.g. "[A] drawback of this model for *our* application is [...]" or "It enables *us* to model [...]"). Issues 1) and 2) may suggest that looking at the sequence of ADU types or linguistic surface features is not enough and a deeper "understanding" and/or domain knowledge are required, especially since the training data is very limited. Lauscher et al. (2018c); Accuosto and Saggion (2019) analyse the impact of SAM to related tasks, suggesting to train on these may mitigate this issue. Finally, issue 2) may be improved by using a joint ADUR+ARE model or an ADUR model that allows to predict non-contiguous spans. Note that we tackle ADU detection in fact with both models in combination because we require the PARTS OF SAME predictions to merge the respective ADUs. This poses a challenge for both models: The ADUR model is trained to predict incomplete instances and the ARE model needs to handle instances from conceptional different types of classes, i.e. argumentative and structural relations.

ARE Error Analysis. For the relation extraction subtask, the general performance is higher than for ADUR with approximately only one third of false negatives or false positives with respect to true positive. However, the performance for CONTRADICTS is much lower than for SUPPORTS, see Figure 5. On reason appears to be the class imbalance. There are substantially less training instances for that class (ratio of $1:8$, see Figure 1). Furthermore, the model significantly overpredicts the CONTRADICTS relations (see confusion matrix in Figure 3). To unravel this phenomenon, we manually analysed 255 relation candidates from different error categories (true positives, false positives, and false negatives). This revealed, that most of the instances falsely predicted as CONTRADICTS can be associated with specific linguistic surface features, especially occurrences of discourse connectors like “however” that are commonly used to express contrastive ideas, but not in this case (see the example in the end of Section 2). Apparently, the model overfits on these shallow markers which is further supported by the fact that all analysed correctly predicted relation instances of that type could be associated with entries of a small set of connectors.[6]

Regarding the SUPPORTS relation, the analysis revealed that sentence boundaries seem to be a very strong signal. An over-proportional amount (85%) of correct predictions has both arguments in the same sentence compared to 20% and 15% for false positives and false negatives, respectively. This is even stronger when taking the argument types into account: SUPPORTS relations that are in the same sentence and connect a DATA ADU with any claim ADU make up for 88% true positives, but only for 19% and 12% of false positives and false negatives. Note, that per definition of the Sci-Arg annotation scheme[7] DATA never participates in a CONTRAST relation which may be one reason why

[6] Consisting of (in decreasing order of frequency): “however”, “but”, “while”, “in contrast”, “though”, “despite”, and “even though”.

[7] The original annotation guidelines can be found here: http://data.dws.informatik.uni-mannheim.de/sci-arg/annotation_guidelines.pdf

relation classification performance is so high. More detailed results of the manual analysis can be found in Figure 5 in the Appendix.

During our analysis we noticed a reasonable amount of potentially mislabeled relation instances (16%), especially missing support relations between OWN CLAIMs. Table 6 shows some examples where relations were correctly detected by the ARE model, but they do not exist in the gold data.

5.3 Ablation Study

We analysed the effect of our approach to add reversed relations. We trained another set of models in a 5-fold cross validation setting with same hyperparameters, but without the augmentation. The resulting mean bootstrapped micro F1 is 0.601, significantly lower than the mean result with augmentation enabled which is 0.762 with $p <$ 1e-10. We gather bootstrapped scores by randomly sampling 10 test document sections, calculate the scores for both model variants as usual and repeat that process for 100 times. Note that there are 114 document sections in total after preprocessing the test set.

6 Related Work

AM is intensively studied for domains like public debates, essays, or legal texts (Lawrence and Reed, 2019). As one of the earliest work for the scientific domain, Teufel and Moens (1999) proposed Argumentative Zoning (AZ) where sentences are classified as AIM, CONTRAST, TEXTUAL, OWN, BACKGROUND, BASIS, or OTHER. The authors created a corpus of 80 annotated full-text papers. They trained Naive-Bayes (NB) and Support-Vector-Machine (SVM) models with hand crafted features and achieved a performance of 0.442 macro-F1. Later work defines similar concepts like "zone of conceptualization" (Liakata, 2010) with classes like EXPERIMENT, BACKGROUND, or MODEL, and trained CRF based models on that (Liakata et al., 2012) (0.18 to 0.76 F1 depending on classes). Guo et al. (2010) compares these schemes with abstract section name detection and trains NB and SVM models. Dasigi et al. (2017) studied the problem of scientific discourse parsing and annotated the result sections of 75 papers with a seven label taxonomy described in de Waard and Pander Maat (2012) like GOAL, FACT, or HYPOTHESIS. They use an LSTM based model augmented with Attention (Vaswani et al., 2017) to obtain sentence representations and present 0.74 F1 performance. In their follow-up work (Li et al., 2021) they achieve a strong 0.841 F1 by using a combination of transfer learning from discourse annotated abstracts (PubMedRCT, Dernoncourt and Lee (2017)) and a model consisting of SciBERT, Attention, BiLSTM, and CRF. In that respect, their approach is similar to ours for ADUR, however, they apply their methods only on the results section of a document and detect full sentence ADUs only. In a similar vein, Achakulvisut et al. (2019) propose a sentence based claim extraction model consisting of BiLSTM and a CRF that they pre-trained on the PubMedRCT dataset. They achieve a performance of 0.790 F1 on a dataset of 1500 abstracts from the Medline dataset. Lauscher et al. (2018a) proposes a tool for automatic ADU recognition and other tasks. Their models are trained on the Sci-Arg dataset and consist of pre-trained word embeddings and a BiLSTM for token classification tasks (e.g. ADUR) and an additional Attention mechanism to obtain sentence representations for the other tasks.

All work mentioned above focuses primarily on the detection and classification of argumentative components. Stab et al. (2014) argues for the need to also analyse argumentative structure, e.g. to automate knowledge base population or reasonable validate claims because that requires to link the respective premises. They also highlight that discourse theory and data is not suited out of the box for argumentative analysis because discourse relations do not cover relevant argumentative relation types and connect primarily neighboring elements which does not reflect argumentative structure. However, Accuosto and Saggion (2019) propose to derive argumentative structure information from discourse data. They annotate a subset of 60 abstracts from the SciDTB scientific discourse dataset (Yang and Li, 2018) with argumentative units and relations. Then, they train models consisting of a BiLSTM, CRF, contextualized word embeddings (ELMo, Peters et al. (2018)) and an encoder pre-trained on the discourse data. They show that adding the encoder significantly improves the performance up to 0.40 F1 argumentative attachment scores, which subsumes argumentative component and relation recognition. Kirschner et al. (2015) created a new corpus by annotating the introduction and discussion sections of 24 scientific articles. The authors consider two argumentative relations, SUPPORT and ATTACK, and also two discourse relations, DETAIL and SEQUENCE borrowed from RST (Mann

Text with ADUs	Annotated	Correction
As the calculations of the wrinkling coefficients are done on a per triangle basis$^{A}_{\text{DATA}}$, the computational time is linear with respect to number of triangles$^{B}_{\text{OWN CLAIM}}$.	$A \leftarrow_S B$	$A \rightarrow_S B$
There are several possibilities to deal with this restriction$^{A}_{\text{OWN CLAIM}}$. One could decide to restrict the simulations to small deformations where the approximation is valid$^{B}_{\text{OWN CLAIM}}$.	-	$A \leftarrow_S B$
As stated in Section 3.3$^{A}_{\text{DATA}}$, two different wrinkle patterns give different wrinkling coefficients for the same triangle geometry$^{B}_{\text{OWN CLAIM}}$. Hence, for the same deformation of the triangle$^{C}_{\text{DATA}}$, corresponding to each pattern, the modulation factors will be different$^{D}_{\text{OWN CLAIM}}$.	$A \rightarrow_S B$ $C \rightarrow_S D$	$A \rightarrow_S B$ $C \rightarrow_S D$ $B \rightarrow_S D$
If a pattern is orthogonal to the deformation direction$^{A}_{\text{OWN CLAIM}}$ (as compared to the other), corresponding modulation factor will be small$^{B}_{\text{OWN CLAIM}}$. In other words, the direction of the deformation favors one pattern over the other$^{C}_{\text{OWN CLAIM}}$.	$A \rightarrow_S B$	$A \rightarrow_S B$ $B \rightarrow_S C$

Table 6: Examples for potentially mislabeled relation instances. $A \rightarrow_S B$ means that the pair of ADUs (A, B) is an instance of the SUPPORTS relation. All proposed corrections are predicted by our model.

and Thompson, 1988), annotated on the sentence level. Recently, Mayer et al. (2020) proposed an argumentation mining pipeline for ADUR and ARE on a new dataset. They annotate 500 Medline abstracts with CLAIM and EVIDENCE ADUs as well as SUPPORT and ATTACK relations. The authors trained and analysed the performance of different models consisting of encoders, like word embeddings, contextualized word embeddings and BERT variants, in combination with a Gated Recurrent Unit (GRU) or LSTM and a CRF. They present a strong micro-F1 of up to 0.92 for ADUR and a performance of up to 0.69 for the full pipeline and conclude that Transformers, especially domain specific ones like SciBERT, work best for SAM at Medline abstracts. Note that, similar to our weak measures, they count predictions as true positive when 75% of the tokens[8] overlap. Another work (Fergadis et al., 2021) that analyses the performance of Transformers for SAM proposes a new corpus of 1000 abstracts with sentence level annotations for CLAIM and EVIDENCE. The authors use a SciBERT applied sentence wise with a BiLSTM over the CLS token embeddings as contextualizer and present a 0.624 macro-F1.

[8]This differs from our weak measures in two ways: Following Lauscher et al. (2018b), we require 50% overlap in means of characters, not tokens.

7 Conclusion and Future Work

In this paper, we presented a pipeline based approach to handle full-text argumentation mining on scientific publications and showed its effectiveness by establishing new state-of-the-art performance on the Sci-Arg corpus. However, there is still a significant gap to human performance. We used PLM based models for both subtasks, argumentative discourse unit recognition (ADUR) and argumentative relation extraction (ARE), and found similar improvements gains (+7%) as reported elsewhere when using Transformers over traditional approaches without Attention mechanism, even without fine-tuning the PLMs.

Our detailed error analysis revealed several findings. First, recognizing instances is much harder than assigning the correct label, which is true for both tasks, but especially for ARE. The performance suffers from shallow processing, i.e. the models are tricked by linguistic surface features like author referencing pronouns in background claims or non-argumentative discourse connectors. Furthermore, ADUR detection struggles a lot in the context of non-contiguous elements which is reasonable because it is trained with incomplete information. This calls for conceptional better modeling of the task, for instance with a joined model for ADUR and ARE. Finally, we could confirm that SAM is a complex problem that is even hard for hu-

mans. However, the low inter-annotator-agreement reported by the Sci-Arg authors and our finding that a significant amount (16%) of the manually analysed ARE instances are questionable labeled raises the need for even more annotation rounds, maybe with multiple domain experts, or a simplified annotation scheme.

Acknowledgments

We would like to thank Aleksandra Gabryszak and the anonymous reviewers for their valuable comments and feedback on the paper. This work has been supported by the German Federal Ministry of Education and Research as part of the projects CORA4NLP (01IW20010) and Software Campus 2.0 (01IS17043).

References

Pablo Accuosto and Horacio Saggion. 2019. Transferring knowledge from discourse to arguments: A case study with scientific abstracts. In *Proceedings of the 6th Workshop on Argument Mining*, pages 41–51, Florence, Italy. Association for Computational Linguistics.

Titipat Achakulvisut, Chandra Bhagavatula, Daniel E. Acuna, and Konrad P. Körding. 2019. Claim extraction in biomedical publications using deep discourse model and transfer learning. *CoRR*, abs/1907.00962.

Khalid Al Khatib, Tirthankar Ghosal, Yufang Hou, Anita de Waard, and Dayne Freitag. 2021. Argument Mining for Scholarly Document Processing: Taking Stock and Looking Ahead. In *Proceedings of the Second Workshop on Scholarly Document Processing*, pages 56–65, Online. Association for Computational Linguistics.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Pradeep Dasigi, Gully A. P. C. Burns, Eduard H. Hovy, and Anita de Waard. 2017. Experiment segmentation in scientific discourse as clause-level structured prediction using recurrent neural networks. *CoRR*, abs/1702.05398.

Anita de Waard and Henk Pander Maat. 2012. Verb form indicates discourse segment type in biological research papers: Experimental evidence. *Journal of English for Academic Purposes*, 11(4):357–366.

Franck Dernoncourt and Ji Young Lee. 2017. PubMed 200k RCT: a dataset for sequential sentence classification in medical abstracts. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 308–313, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Aris Fergadis, Dimitris Pappas, Antonia Karamolegkou, and Haris Papageorgiou. 2021. Argumentation mining in scientific literature for sustainable development. In *Proceedings of the 8th Workshop on Argument Mining*, pages 100–111, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Santo Fortunato, Carl T. Bergstrom, Katy Börner, James A. Evans, Dirk Helbing, Staša Milojević, Alexander M. Petersen, Filippo Radicchi, Roberta Sinatra, Brian Uzzi, Alessandro Vespignani, Ludo Waltman, Dashun Wang, and Albert-László Barabási. 2018. Science of science. *Science*, 359(6379):eaao0185.

Nancy Green. 2014. Towards creation of a corpus for argumentation mining the biomedical genetics research literature. In *Proceedings of the First Workshop on Argumentation Mining*, pages 11–18, Baltimore, Maryland. Association for Computational Linguistics.

Yufan Guo, Anna Korhonen, Maria Liakata, Ilona Silins, Lin Sun, and Ulla Stenius. 2010. Identifying the information structure of scientific abstracts: An investigation of three different schemes. In *Proceedings of the 2010 Workshop on Biomedical Natural Language Processing*, pages 99–107, Uppsala, Sweden. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]*. ArXiv: 1412.6980.

Christian Kirschner, Judith Eckle-Kohler, and Iryna Gurevych. 2015. Linking the thoughts: Analysis of argumentation structures in scientific publications. In *Proceedings of the 2nd Workshop on Argumentation Mining*, pages 1–11, Denver, CO. Association for Computational Linguistics.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001), Williams College, Williamstown, MA, USA, June 28 - July 1, 2001*, pages 282–289. Morgan Kaufmann.

Anne Lauscher, Goran Glavaš, and Kai Eckert. 2018a. ArguminSci: A tool for analyzing argumentation and rhetorical aspects in scientific writing. In *Proceedings of the 5th Workshop on Argument Mining*, pages 22–28, Brussels, Belgium. Association for Computational Linguistics.

Anne Lauscher, Goran Glavaš, and Simone Paolo Ponzetto. 2018b. An argument-annotated corpus of scientific publications. In *Proceedings of the 5th Workshop on Argument Mining*, pages 40–46, Brussels, Belgium. Association for Computational Linguistics.

Anne Lauscher, Goran Glavaš, Simone Paolo Ponzetto, and Kai Eckert. 2018c. Investigating the role of argumentation in the rhetorical analysis of scientific publications with neural multi-task learning models. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3326–3338, Brussels, Belgium. Association for Computational Linguistics.

Anne Lauscher, Goran Glavaš, Simone Paolo Ponzetto, and Kai Eckert. 2018d. Investigating the Role of Argumentation in the Rhetorical Analysis of Scientific Publications with Neural Multi-Task Learning Models. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3326–3338, Brussels, Belgium. Association for Computational Linguistics.

John Lawrence and Chris Reed. 2019. Argument mining: A survey. *Computational Linguistics*, 45(4):765–818.

Xiangci Li, Gully Burns, and Nanyun Peng. 2021. Scientific discourse tagging for evidence extraction. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2550–2562, Online. Association for Computational Linguistics.

Maria Liakata. 2010. Zones of conceptualisation in scientific papers: a window to negative and speculative statements. In *Proceedings of the Workshop on Negation and Speculation in Natural Language Processing*, pages 1–4, Uppsala, Sweden. University of Antwerp.

Maria Liakata, Shyamasree Saha, Simon Dobnik, Colin Batchelor, and Dietrich Rebholz-Schuhmann. 2012. Automatic recognition of conceptualization zones in scientific articles and two life science applications. *Bioinformatics (Oxford, England)*, 28(7):991–1000.

William C. Mann and Sandra A. Thompson. 1988. Rhetorical Structure Theory: Toward a functional theory of text organization. *Text - Interdisciplinary Journal for the Study of Discourse*, 8(3).

Tobias Mayer, Elena Cabrio, and Serena Villata. 2020. Transformer-based argument mining for healthcare applications. In *ECAI 2020 - 24th European Conference on Artificial Intelligence, 29 August-8 September 2020, Santiago de Compostela, Spain, August 29 - September 8, 2020 - Including 10th Conference on Prestigious Applications of Artificial Intelligence (PAIS 2020)*, volume 325 of *Frontiers in Artificial Intelligence and Applications*, pages 2108–2115. IOS Press.

Josh M. Nicholson, Milo Mordaunt, Patrice Lopez, Ashish Uppala, Domenic Rosati, Neves P. Rodrigues, Peter Grabitz, and Sean C. Rife. 2021. Scite: A smart citation index that displays the context of citations and classifies their intent using deep learning. *Quantitative Science Studies*, 2(3):882–898.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Reuven Rubinstein. 1999. The Cross-Entropy Method for Combinatorial and Continuous Optimization. *Methodology And Computing In Applied Probability*, 1(2):127–190.

M. Schuster and K.K. Paliwal. Nov./1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

Christian Stab, Christian Kirschner, Judith Eckle-Kohler, and Iryna Gurevych. 2014. Argumentation mining in persuasive essays and scientific articles from the discourse structure perspective. In *Proceedings of the Workshop on Frontiers and Connections between Argumentation Theory and Natural Language Processing, Forlì-Cesena, Italy, July 21-25, 2014*, volume 1341 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Manfred Stede and Jodi Schneider. 2018. Argumentation Mining. *Synthesis Lectures on Human Language Technologies*, 11(2):1–191.

Simone Teufel and Marc Moens. 1999. Discourse-level argumentation in scientific articles: human and automatic annotation. In *Towards Standards and Tools for Discourse Tagging*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. *arXiv:1706.03762 [cs]*. ArXiv: 1706.03762.

A. Viterbi. 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13(2):260–269. Conference Name: IEEE Transactions on Information Theory.

Douglas Walton. 2001. *Informal Logic: A Pragmatic Approach*, 2 edition. Cambridge University Press.

An Yang and Sujian Li. 2018. SciDTB: Discourse dependency TreeBank for scientific abstracts. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 444–449, Melbourne, Australia. Association for Computational Linguistics.

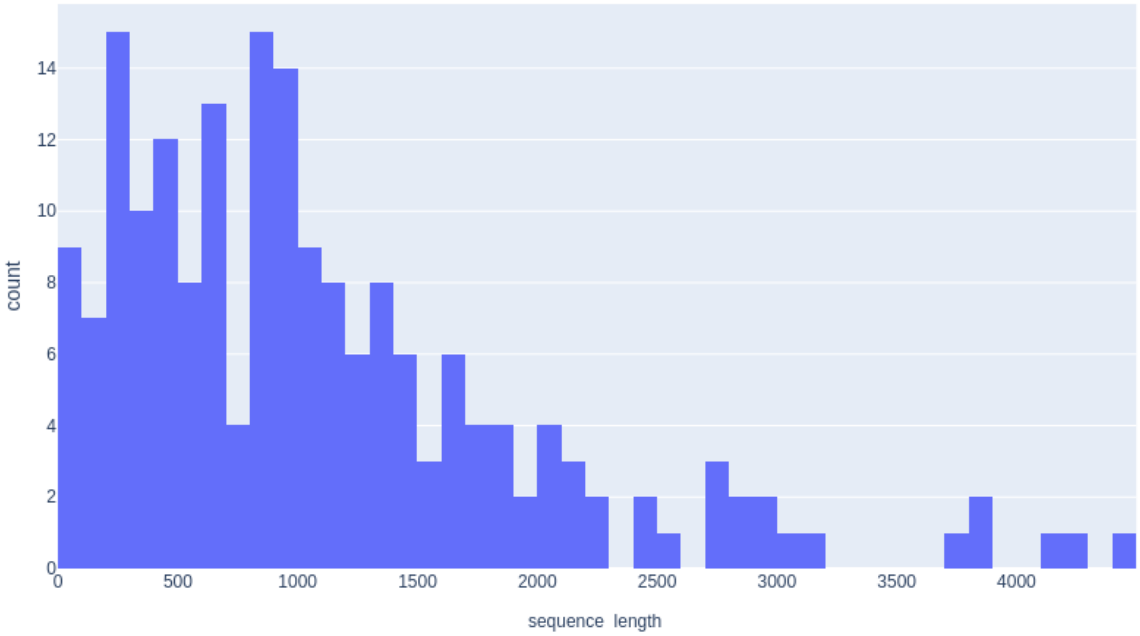


Figure 4: **Distribution of input sequence lengths.** This is after splitting the document text into sections and tokenization. Note that this is primarily relevant for the ADU model since we use a much smaller token window size k to restrict the input for the ARE model.

A Appendix

A.1 Preprocessing

We use the following regular expression pattern to match content in the beginning of the files that we remove: "`<\?xml[^>]*>[^<]*<Document xmlns:gate="http://www.gate.ac.uk"[^>]*>[^<]*`" (without the outer quotes). Main sections are marked by `<h1>SECTION_HEADING</h1>` in the Sci-Arg corpus where `SECTION_HEADING` is any text, so we use this regular expression pattern to split the texts: "`<H1>`"(without the quotes). Note, that we keep that content in the input. The input sequence lengths for the ADU model reaches still values > 4000. Figure 4 shows its distribution.

A.2 Experimental Setup and Hyperparameters

We use the AllenNLP framework to implement the models and execute the training. As PLM, we use the uncased variant of SciBERT (Beltagy et al., 2019) as provided by AllenAI[9]. ADAM (Kingma and Ba, 2014) is used as optimizer. We use batch sizes of 8 and 128 for ADU recognition and RE, respectively, that are derived from resource constraints. The ADU tags are encoded with the BIOUL tagging scheme. For the RE subtask, we hand-picked embedding sizes of 13 and 3 for the ADU-tags and argument-tags, respectively, that are derived form the number of classes.[10]

As a result of the hyperparameter search, we use the following parameters for the ADU recognition task: a learning rate of 0.005, dropout probability of 0.5 before and after the PLM and 0.4394 in the LSTM, a gradient normalization threshold of 7.0, a patience of 20 epochs for early stopping, two layers for the LSTM with a hidden size of 300. In the case of RE, we got the following values: a learning rate of 0.0005, a dropout probability of 0.3061 before and after the PLM and 0.4394 in the LSTM, a gradient normalization threshold of 4.12, 4 layers for the LSTM with a hidden size of 430, 193 filters for the CNN (with ngram sizes of 3, 5, 7, and 10), a hidden size of 860 for the final projection layer, a token window size `k` of 479 tokens around the center of the candidate argument pair, a max inner token distance `d` between the arguments of 177[11], and finally, we use a factor

[9] see `https://huggingface.co/allenai/scibert_scivocab_uncased`

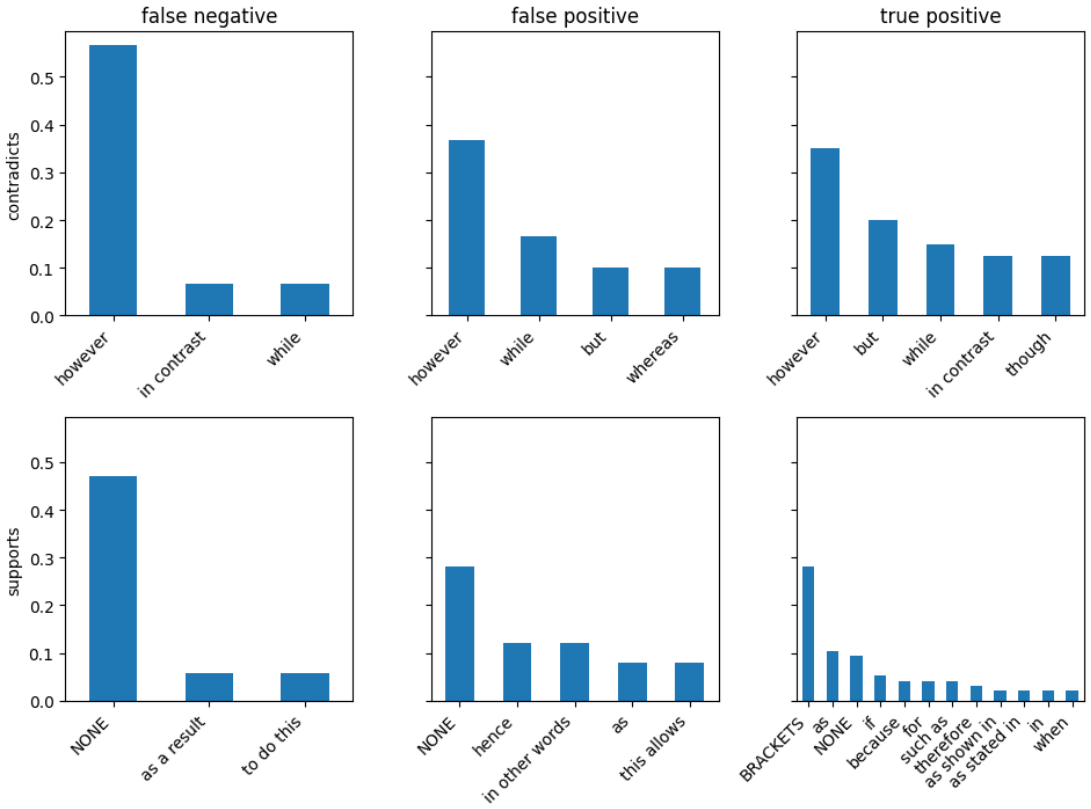
[10] Note, the three ADU-tags are each BIOUL encoded and the argument types, *head* and *tail*, are BIO encoded.

[11] This causes a loss of 0.23% of SUPPORT instances and 0.5% of PARTS OF SAME instances, which is neglectable.

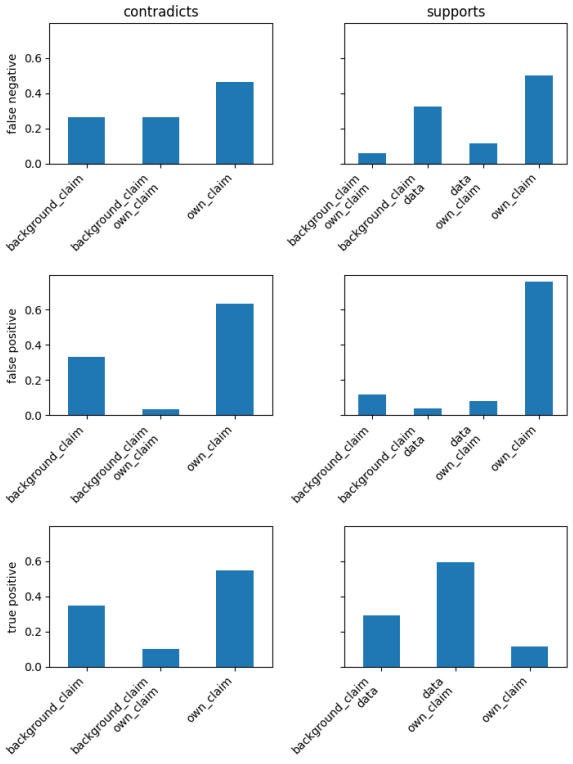
of three for the amount of negative examples, i.e. we add three times as many existing argumentative ADU pairs as NO RELATION instances which we sample from all available pairs without a relation label and within the distance constraint.

A.3 Training Resources

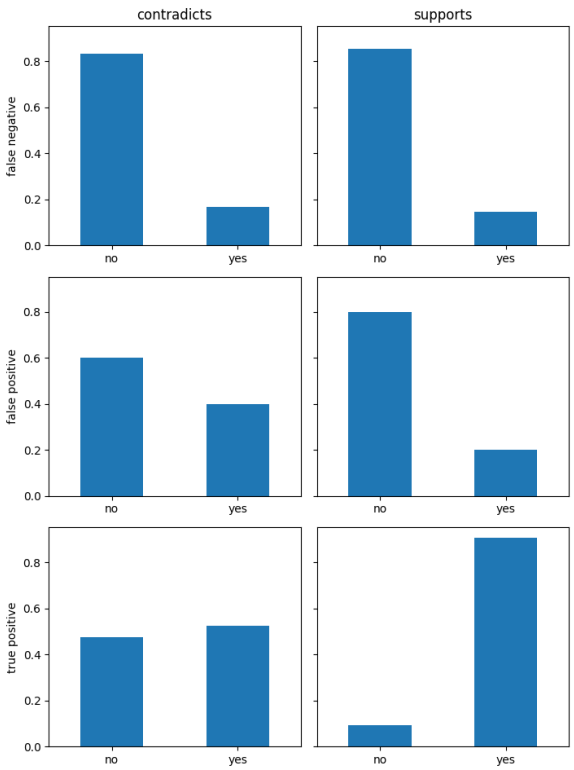
The hyperparameter search was performed on a single Nvidia RTX A6000 (48GB). The training of the final models, i.e. 5 for each subtask, and inference was calculated on single Nvidia GeForce GTX 1080 Ti (12GB). The total training time for all final models was 5h51m for ADUR and 40h17m for ARE.



(a) Distribution of **connecting phrases**. Despite being no real discourse connectors, we also collected markers like BRACKETS that seem to be important surface features. NONE indicates that no connective element was found.



(b) Distribution of **relation arguments** (sorted and mentioned only once if both arguments are the same).



(c) Distribution of the feature that both arguments are **in the same sentence**.

Figure 5: Results of the manual error analysis for argumentative relation extraction. The figures show proportions of different features (connectors, arguments, and same sentence feature) at different subsets by error type (false negative, false positive, or true positive). The lowest entries per category are excluded. Values are calculated on a manually collected subset of 255 relation instances in total.