ARIES: A General Benchmark for Argument Relation Identification

Debela Gemechu, Ramon Ruiz-Dolz and Chris Reed
Centre for Argument Technology (ARG-tech)
University of Dundee
Dundee DD1 4HN, United Kingdom

Abstract

Measuring advances in argument mining is one of the main challenges in the area. Different theories of argument, heterogeneous annotations, and a varied set of argumentation domains make it difficult to contextualise and understand the results reported in different work from a general perspective. In this paper, we present ARIES, a general benchmark for Argument Relation Identification aimed at providing a standard evaluation for argument mining research. We evaluated three different architectures for Argument Relation Identification on ARIES: sequence and token modelling, and sequence-to-sequence alignment, together with the three main Transformer-based model architectures: encoder-only, decoder-only, and encoder-decoder. Furthermore, the benchmark consists of eight different argument mining datasets, covering the most common argumentation domains, and standardised with the same annotation structures. This paper provides a first comprehensive and comparative set of results in argument mining across a broad range of configurations to compare with, both advancing the state-of-the-art, and establishing a standard way to measure future advances in the area. Across varied task setups and architectures, our experiments reveal consistent challenges in cross-dataset evaluation, with notably poor results. Given the models' struggle to acquire transferable skills, the task remains challenging, opening avenues for future research.

1 Introduction

Argument mining was originally defined as the task of automatically identifying argument structures from unstructured natural language inputs (Mochales and Moens, 2011). Although argument mining research has been split into several subtasks in the literature such as segmentation, argument classification and argument relation identification (Lippi and Torroni, 2016; Lawrence and Reed, 2020) it is the latter that represents the main challenge in argument mining due to its complexity. Argument Relation Identification (ARI) starts from the point where all the relevant argument sequences have been segmented, and its main objective is to identify argumentative relations between them building complete argumentative structures. Therefore, it is the ARI subtask that provides us with the argument structures from unstructured natural language. In addition, while outstanding results have been reported in the previous stages, results on ARI are more limited, representing one of the most difficult tasks in natural language processing due to its implicitness, the lack of data, and the lack of solid baselines with which to compare.

One of the main challenges in the area of argument mining, however, has always been to compare advances in different contexts, understanding these contexts as different annotation theories or argumentative domains. Therefore, previous work reports different findings and advances, but it does that without providing a general picture of them and a comprehensive understanding of their findings for the argument mining community as a whole. A lack of a consistent cross-domain benchmark, as it has been done in many other areas of natural language processing (see GLUE (Wang et al., 2018), Superglue (Wang et al., 2019), TweetEval (Barbieri et al., 2020), or Superb (Yang et al., 2021) among others), hinders our advances as a research community. Although previous effort has been put in this direction, none of these previous work considers state-of-the-art NLP algorithms considering multiple language modelling approaches, and typically focused on specific tasks or domains (Cocarascu et al., 2020; Ruosch et al., 2022). Providing relevant results in the good direction, but limited in terms of generalisability. Therefore, the definition of a general benchmark for state-of-the-art argument mining is something that remains unaddressed. This limitation, taking the success of the GLUE benchmark for natural language under-

Proceedings of the 11th Workshop on Argument Mining, pages 1–14
August 15, 2024 ©2024 Association for Computational Linguistics

standing tasks (Wang et al., 2018) into account, motivates the development of an argument mining-specific benchmark to comprehensively evaluate and measure the advances done in the area.

In this paper, we present the Argument Relation Identification Evaluation Strategy (ARIES), a robust, cross-dataset benchmark for evaluating existing and future contributions to the ARI task. ARIES represents the first and most extensive benchmark to evaluate ARI systems, thus providing a robust framework for comparative evaluation of argument mining systems. Our main contribution is the formal definition of ARIES, including eight different corpora, three different natural language modelling approaches, and three different model architectures. Furthermore, we carried out extensive experimentation, implementing the previous natural language modelling approaches and model architectures with different pre-trained language models. From our results, we do not only advance the state-of-the-art in ARI, but also identify a concerning limitation of the generalisation capabilities of argument mining systems. This way, ARIES provides an ideal base on which to compare, propose, and implement argument mining systems addressing the ARI task.

2 Related Work

The latest advances in natural language processing have been reflected in argument mining and especially in ARI research, the most challenging part of it. These natural language processing advances have been gradually integrated into argument mining systems with (in order) the use of LSTM networks (Cocarascu and Toni, 2017), the Transformer architecture (Ruiz-Dolz et al., 2021a), contrastive learning (Shi et al., 2022), generative language models (Bao et al., 2022), end-to-end architectures (Morio et al., 2022), or the most recent large language models (LLMs) (Gorur et al., 2024). All these advances, however, are difficult to compare and contextualise due to a lack of a standardised set of evaluation baselines.

Some effort has been put with previous research on the definition of benchmarks for argument mining. Initially proposed in (Cabrio and Villata, 2014), the authors define NoDE, a natural language argument benchmark consisting of three datasets and 792 related proposition pairs. In this early benchmark, the authors pointed out the needs of standardising the evaluation of argument mining systems. Following this direction, Cocarascu et al. (2020) extended the previous benchmark with a total of ten datasets containing 35,918 related proposition pairs. Both benchmarks exclusively focused on the classification of argument relations, a subset of the ARI task in which the relation is assumed to be known, limiting their applicability in more general situations. Recently in (Ruosch et al., 2022), the authors address this limitation by proposing a benchmark for argument mining (BAM), in which all the argument mining subtasks are brought into consideration together. The BAM framework, however, is proposed as a pipeline-like method combining different previously existing argument mining systems to cover the complete argument mining process (Ruosch et al., 2023), rather than a thorough analysis of state-of-the-art NLP modelling techniques and architectures. Furthermore, its current version only contains argumentative information in scientific documents, making it a valuable resource for this domain but limiting its generalisability to other application domains.

3 Benchmark

The main contribution of this paper is the definition of ARIES, a state-of-the-art benchmark for argument relation identification in datasets of different domain and nature, which can be used as a reference to advance and to relativise the real impact of new findings in this area. Furthermore, ARIES also reflects on a wide variety of model architectures, providing more insight on the capabilities of state-of-the-art algorithms. This section provides an in-depth presentation of all the variables taken into account in the proposed ARIES benchmark.

3.1 Data

In order to develop a robust, challenging and wide-ranging assessment, we include eight different datasets as part of ARIES. These eight datasets were selected mostly based on two criteria. First, we selected the most representative datasets on the area of argument mining. This way, ARIES can be used as reference, not only for future contributions, but also for these ones already existing. Second, our selection was determined by our goal of creating a sufficiently heterogeneous dataset in terms of domain to be able to measure the robustness of state-of-the art systems. Therefore, ARIES consists of eight different argumentation domains. The eight datasets included into ARIES are: MTC (Peldszus and Stede, 2015), AAEC (Stab and Gurevych,

Dataset	Domain	Inferences	Conflicts	Neutral	Total
MTC	Structured Argumentation	272	108	713	1,093
AAEC	Essay	4,841	497	10,676	16,014
CDCP	Financial	694	82	1,552	2,328
ACSP	Scientific	8,069	697	17,532	26,298
AMP	Online	2,111*	-	5,929	8,040
ABSTRCT	Medical	2,290	344	4,581	7,215
US2016	Political	2,765	866	7,262	10,893
QT30	Question Answering	2,714	545	6,518	9,777
Total	-	23,756	3,139	54,763	81,658

Table 1: Summary of the ARI datasets included in the ARIES benchmark. We use * to indicate that AMP involves only binary labels, indicating whether a relation is present or not.

2017), CDCP (Park and Cardie, 2018), ACSP (Lauscher et al., 2018), AMP (AMPERSAND) (Chakrabarty et al., 2019), ABSTRCT (Mayer et al., 2020), US2016 (Visser et al., 2020), and QT30 (Hautli-Janisz et al., 2022). A summary of the most relevant features of these eight corpora is depicted in Table 1.

3.2 Task

The ARIES benchmark evaluates the ARI task. ARI consists of the identification of existing argumentative relations between two or more Argumentative Discourse Units (ADUs). This way, this task takes an unstructured set of ADUs as its input and outputs complete structured arguments, making it the cornerstone of argument mining. For practicalities, within the ARIES framework, we define the ARI task as a three-class classification problem, considering the Inference, Conflict, and Neutral classes of argumentative relations. The inference relation represents an argumentative support, the conflict relation represents an argumentative attack, and the neutral class indicates that there exists no argumentative relation between a set of ADUs. Some models distinguish a fourth category of rephrase because it has become increasingly clear in linguistic work (Koszowy et al., 2022) that this is relation is a key driver of argumentation. It is, however, far from ubiquitous in argument mining research, and so is not adopted in the current ARIES framework.

It is also important to emphasise that the ARIES benchmark goes one step further compared to the Argument Relation Classification (ARC) task, which only considers attacks and supports. While ARC can be framed as a sentiment analysis problem with positive and negative sentiments, and it is based on the assumption that the existing argument relations are all known, ARI does not make such assumption, and therefore modelling the underlying (and sometimes implicit) argumentative features of ADUs is essential if we want an algorithm to succeed on this task.

3.3 Models

We consider three different natural language modelling approaches in the ARIES benchmark: sequence classification (see Figure 1a), token classification (see Figure 1b), and sequence-to-sequence alignment (see Figure 1c). Complementing these modelling approaches, we have also included three different model architectures in our benchmark: encoder only, decoder only, and encoder-decoder. This way, we cover the majority of the existing approaches for argument mining in the literature.

3.3.1 Sequence Classification

First, our benchmark reports results when addressing ARI as a sequence (pair) classification task. This way, our sequence classification benchmark models the conditional probability of the most likely relation class (i.e., $\hat{s}$) for a given pair of ADUs as depicted in Equation 1.

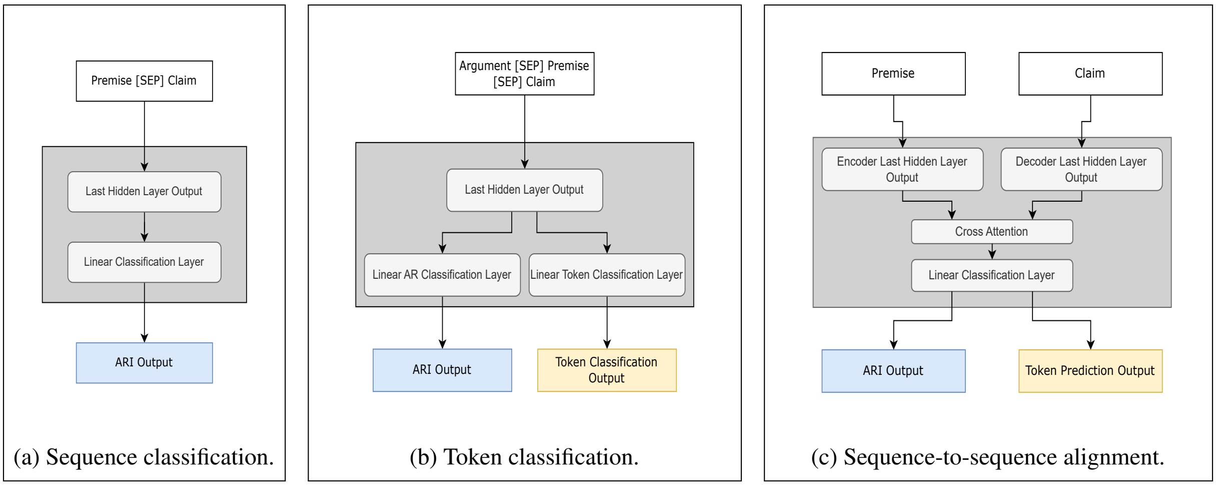
$$\hat{s} = \arg\max_{s \in S} P(s | x_1^N, y_1^M) \quad (1)$$

where S stands for the complete set of possible argumentative relations (i.e., Inference, Conflict, or Neutral), x_1^N represents the first ADU of length N, and y_1^M is the second ADU of length M. With this framing, the two ADU inputs are treated as a whole sequence of text, modelling natural language at a higher level and looking for sequence features that can be helpful to determine whether a pair of propositions is related with an inference, a conflict, or presents no relation between them. This is one of the most widely researched approaches when it comes to the identification of argument relations (Cocarascu and Toni, 2017; Ruiz-Dolz et al., 2021a; Shi et al., 2022; Kikteva et al., 2023; Gorur et al., 2024).

The loss ($\mathcal{L}_{\text{class}}$) for the argument relation classification task is computed using the standard cross-entropy loss based on the predicted logits and true labels for the argument relation type.

$$\mathcal{L}_{\text{class}} = -\frac{1}{B} \sum_{i=1}^{B} \log(\text{softmax}(r_i^{pred})) \cdot r_i^{true} \quad (2)$$

where r_i^{true} represents the true label of the argument relation type for the i-th sample and r_i^{pred}


Premise [SEP] Claim
Last Hidden Layer Output
Linear Classification Layer
ARI Output
Argument [SEP] Premise [SEP] Claim
Last Hidden Layer Output
Linear AR Classification Layer
Linear Token Classification Layer
ARI Output
Token Classification Output
Premise
Claim
Encoder Last Hidden Layer Output
Decoder Last Hidden Layer Output
Cross Attention
Linear Classification Layer
ARI Output
Token Prediction Output

(a) Sequence classification.

(b) Token classification.

(c) Sequence-to-sequence alignment.

Figure 1: The architecture for the three tasks: (a) sequence classification, (b) token classification, and (c) sequence-to-sequence alignment. In this paper, we evaluate only the ARI output (highlighted in light blue), while the token classification and token prediction outputs (highlighted in light yellow) serve as auxiliary tasks and are not evaluated.

denotes the predicted logits of the argument relation type for the i-th sample.

3.3.2 Token Classification

Second, we also benchmark ARI as a token classification task, aiming to predict the span of conclusions given a premise or vice versa, while simultaneously predicting the argument relation between the premise and conclusion in a multi-task setup. Inspired by Eger et al. (2017), who modeled argument mining as a token classification task, jointly addressing component identification and relation identification, assigning each token a label indicating the category of the component and the argument relation type, our work acknowledges their finding of sub-optimal coupling between the two tasks and advocates for treating them separately while modelling them jointly. Consequently, we adopt a multi-task setting that independently models both tasks. The multi-task learning setup encompasses two primary objectives: span prediction and argument relation identification.

For the span prediction sub-task, given an argument (i.e., the complete structure resulting from the concatenation of the premises and conclusion), we model the boundaries of the conclusion within the argument given the premises and vice-versa (See Appendix A.2 for more details regarding the input format). Our token classification approach, therefore, first models the conditional probability of the most likely span boundaries (i.e., $\hat{t}$) as depicted in Equation 3.

$$\hat{t} = \arg\max_{t \in T} P(t|n_1, ..., n_{i-1}, n_i) \qquad (3)$$

where T represents the set of possible token labels (i.e., beginning (B), inside (I), or outside (O)), and n_i represents each token at a given position i, followed by the previously observed tokens in the complete argument sequence. This approach treats tokens in a more independent way than in sequence classification, allowing to look for lower level features, where each token is assigned a specific label. The loss ($\mathcal{L}_s$) for the span prediction sub-task is computed using the standard cross-entropy loss based on the predicted and true labels for each token in the argument.

$$\mathcal{L}_{span} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} Y_{p,ij}^{true} \cdot \log(Y_{p,ij}^{pred}) \qquad (4)$$

where N represents the total number of tokens in the argument, C is the set of token labels, $Y_{p,ij}^{true}$ denotes the ground truth probability of token i belonging to class j, and $Y_{p,ij}^{pred}$ is the predicted probability of token i belonging to class j.

The second step in the token classification approach is the identification of argumentative relations. This second task involves predicting the argument relation between the premise and conclusion in a similar way as described in Equation 1. It takes both the premise and conclusion resulting from the previous span detection sub-task as inputs and predicts the argument relation. Since the primary focus is on ARI, the span detection serves as an auxiliary task. The loss from the token classification approach results from adding the previously defined span prediction loss $\mathcal{L}_{span}$ to the sequence classification loss $\mathcal{L}_{class}$ (see Equation 2), resulting in the overall loss ($\mathcal{L}_{total}$) as defined in Equation 5.

$$\mathcal{L}_{total} = \mathcal{L}_{span} + \mathcal{L}_{class} \tag{5}$$

3.3.3 Sequence-to-sequence Alignment

Finally, the third approach included in the ARIES benchmark corresponds to a sequence-to-sequence alignment modelling of the relation between argument proposition pairs. In this last approach, we address ARI in a similar way as machine translation is done, where the model is trained to predict a complete sequence related to the input (Stahlberg, 2020). Therefore, we consider the argument premise as the input and provide the argument claim as the expected output, modelling this way the semantic connections between both propositions resulting in the argumentative relation between premise and claim as depicted in Equation 6.

$$\hat{c}_1^N = \underset{c_1^N}{\arg\max}\, P(c_1^N | p_1^M) \tag{6}$$

Where c_1^N stands for the output claim sequence of length N, and p_1^M for the input premise sequence of length M. The sequence-to-sequence alignment approach is divided into two steps. First, we do the sequence-to-sequence modelling according to Equation 6 attempting to improve the embedding representation of our argumentative inputs (i.e., premise-claim pairs). Second, we leverage the embedding of the premise-claim representations to train a classifier that predicts our three relation classes in a similar way as described in Equation 1. Although less researched in the literature, sequence-to-sequence approaches have also been recently investigated in the area of argument mining thus making them an important addition to our global benchmark (Kawarada et al., 2024).

The loss ($\mathcal{L}_{\text{seq}}$) for the sequence-to-sequence alignment approach is computed using the standard cross-entropy loss based on the predicted logits and true labels for each token in the conclusion sequence.

$$\mathcal{L}_{\text{seq}} = -\frac{1}{B}\sum_{i=1}^{B}\sum_{j=1}^{N}\log(\text{softmax}(c_{ij}^{pred})) \cdot c_{ij}^{true} \tag{7}$$

Where B indicates the batch size, N is the length of the conclusion sequence, c_{ij}^{true} denotes the true label of the j-th token in the i-th sample, c_{ij}^{pred} represents the predicted logits of the j-th token in the i-th sample, and softmax$(\cdot)$ represents the output of the softmax function.

The loss ($\mathcal{L}_{\text{class}}$) for argument relation classification is computed using the same loss function as in the sequence classification approach defined in Equation 2. The overall loss ($\mathcal{L}$) is the sum of both losses:

$$\mathcal{L} = \mathcal{L}_{\text{seq}} + \mathcal{L}_{\text{class}} \tag{8}$$

This hybrid approach, combining sequence-to-sequence modelling with ARI, allows us to capture the relationship between the premise and conclusion while effectively predicting argument relation types.

3.3.4 Model Architectures

In addition to the three natural language modelling approaches, we have also included the three main model architectures in state-of-the-art natural language processing. This way, we consider encoder-only (Devlin et al., 2019), decoder-only (Brown et al., 2020), and encoder-decoder (Vaswani et al., 2017) architectures. For the first two natural language modelling approaches (i.e., sequence and token classification), the ARIES benchmark considers the three possible architectures. However, for the sequence-to-sequence alignment approach, we can only rely on the encoder-decoder architecture, given its nature requiring both encoder and decoder (see Appendix A.2 for more details).

4 Experiments

4.1 Experimental Setup

We use Adam optimisation (Kingma and Ba, 2014) to minimise the loss function, using a learning rate of 2×10^{-5} and categorical cross-entropy loss and a batch size of 16 (more details on the experimental setup is provided in Appendix A). The dataset is randomly partitioned, with 70%, 10%, and 20% allocation for training, validation, and testing respectively, ensuring uniformity throughout the dataset. Refer to Table 1 for the breakdown of argument relations. All our results represent the average of three runs using different random seeds. Precision, recall, and F1-score are computed, and macro-averaged F1-scores are reported for the test dataset. The code used

Task	**Architecture**	**Model**	**Eval**				**Dataset**				
				MTC	AAEC	CDCP	ACSP	AMP	AbstRCT	US2016	QT30
SeqCls	**ED**	**RoBERTa**	ID	63	75	**72**	82	**84***	**84**	76	83
			CD	35	47	40	**50**	51*	40	48	45
	DD	**DialoGPT**	ID	66	**78**	**72**	**84**	**84***	82	**79**	**85**
			CD	40	**48**	41	49	**52***	39	**49**	**49**
	ED-DD	**T5**	ID	65	74	71	83	80*	80	74	84
			CD	37	37	36	38	40*	37	38	39
TokCls	**ED**	**RoBERTa**	ID	61	76	68	80	-	81	73	82
			CD	33	42	31	37	-	33	37	35
	DD	**DialoGPT**	ID	63	77	70	82	-	80	75	82
			CD	34	42	33	39	-	35	38	37
	ED-DD	**T5**	ID	62	73	65	81	-	78	71	80
			CD	33	35	34	36	-	33	35	33
SeqAln	**ED-DD**	**T5**	ID	**68**	75	70	81	78*	83	76	83
			CD	**41**	42	**42**	42	46*	**41**	43	41

Table 2: In-dataset (ID) and cross-dataset (CD) macro-averaged F1-score results. We use * to denote that the evaluation results reported on the AMP represent binary predictions.

in our experiments can be publicly accessed at https://github.com/debelatesfaye/ArgumentMining24-ARIES-Benchmark.

4.2 Evaluation Setup

In-Dataset Evaluation. In the in-dataset (ID) evaluation, each model configuration is trained and evaluated on the same dataset, enabling us to assess the performance of models within the same domain. Each of the three task setups: sequence classification (SeqCls), token classification (TokCls), and sequence-to-sequence alignment (SeqAln), are trained and evaluated across the datasets. The three task setups are evaluated on eight datasets, with the exception of TokCls, which is evaluated on all datasets except AMP. This exception arises because AMP solely focuses on the pair of propositions, while TokCls requires the entire argument in addition to the pair of propositions. Within SeqCls and TokCls, the three transformer architectures—Encoder only (ED), Decoder only (DD), and Encoder-Decoder (ED-DD)—are evaluated. However, considering the specific requirements of the SeqAlg task and its architectural demands, only the ED-DD configuration is evaluated. This provides three architecture variants for the SeqCls task: ED-based SeqCls, DD-based SeqCls, and ED-DD-based SeqCls, each of which undergoes training and evaluation across eight datasets, providing a total of 24 configurations, respectively. The TokCls task encompasses 21 configurations across the seven datasets, whereas the Seq-Alg task is limited to the ED-DD configuration across eight datasets, totaling eight configurations.

Cross-Dataset Evaluation. The cross-dataset (CD) evaluation setup involves training each model on one dataset and subsequently evaluating its performance on each of the remaining seven datasets, providing insights into their generalisation and domain adaptability. Accordingly, for both the SeqCls and TokCls tasks, the three transformer architectures are trained on eight and seven training datasets, respectively, resulting in a total of 45 models (24 for SeqCls and 21 for TokCls tasks). Subsequently, each model is evaluated on the remaining datasets not used for training, resulting in an evaluation matrix encompassing a total of 294 configurations (3 architectures * 8 training datasets * 7 evaluation datasets for SeqCls and 3 architectures * 7 training datasets * 6 evaluation datasets for TokCls). Conversely, the Seq-Alg task, involves training the ED-DD configuration across the eight training datasets and evaluated on the remaining 7 dataset not used for training, resulting in an evaluation matrix encompassing 56 configurations (1 architecture * 8 training datasets * 7 evaluation datasets).

4.3 Results

Table 2 illustrates the macro F1-scores achieved in both ID and CD evaluation scenarios. In the ID evaluation configuration, the F1-scores are determined from models trained and evaluated on the same dataset. Conversely, in the CD evaluation setup, each model undergoes training on one dataset, and evaluated on the remaining datasets. The average F1-scores of each model trained on one dataset and evaluated on the remaining is reported in Table 2. The comprehensive CD evaluation results can be found in Tables 4 and 5, located in the Appendix.

As can be seen from Table 2, in the ID evaluation, an average F1-score of 76.1% was achieved across the three tasks and architectures. Conversely, in the CD evaluation, the average F1-score was notably lower at 42.7%. To facilitate a direct comparison among the three task setups, we calculate the average performance across the entire dataset, excluding AMP, as TokCls does not undergo evaluation on this specific dataset. Accordingly, the sequence-classification setup attains an average F1-score of 77.5%, 40.2% across all architectures in both ID and CD evaluations, respectively, while token-classification models achieve an average F1-score of 74.3%, 35.5%. Sequence-to-sequence alignment task achieves comparable performance with sequence classification task with an average F1-score of 76.8% and 40.4% on ID and CD setups, respectively. It is noteworthy that the average F1-score of the ED-DD architecture for sequence classification and token classification is 34.7% in CD evaluations. This represents a 5.7% improvement in the task performance in the CD evaluation setup. This observed gain underscores the task's effectiveness, especially when compared to the lower performance achieved by the same underlying model (T5) in the other two task setups, highlighting its ability to learn transferable features across domains. This phenomenon could be attributed to the inherent nature of the task setup, which presents challenging learning scenarios, potentially combating shortcut learning and encouraging the model to learn more generalised representations. Moreover, it might also suggests the task allowing to learn the alignment of the premise and conclusion based on the outputs of the encoder and decoder.

Sequence classification models exhibited faster convergence during training compared to token classification and sequence-to-sequence alignment counterparts, indicating their ability to learn and adapt more efficiently to the task at hand. The slower convergence observed in token classification and sequence-to-sequence alignment setups could be attributed to the complexity of the tasks, requiring the model to learn token-level relationships to predict argument relations.

Across all dataset and task combinations, ED (RoBERTa) configurations demonstrate an average F1-score of 75.4%, 38.4% in ID and CD evaluation settings, respectively. In contrast, DD configurations achieve an F1-score of 76.8%, 39% in the ID and CD evaluation settings, respectively. Configurations using DialoGPT exhibit a 1.4% improvement over RoBERTa across datasets and tasks in ID evaluation. DialoGPT's superior performance could be attributed to its pre-training strategy and dataset, which specifically target dialogical datasets extracted from Reddit comment chains. As DialoGPT is exclusively pre-trained on dialogical data, configuration utilising the model could leverage the argument-relevant features encoded during its pre-training stage. This advantage might enable DialoGPT based configurations to outperform configurations based on models pre-trained on generic datasets. The specificity of DialoGPT's pre-training strategy likely helps capture the subtleties of argumentation and discourse, thereby enhancing performance in ARI tasks.

Moreover, the performance variations among the transformers architectures can be indicative of the relevance of the underlying pre-training objectives and architectures to ARI. Notably, the next sentence prediction objective, crafted for classification tasks involving sequence pairs, aligns with ARI, as the task involves pairs of propositions. However, RoBERTa, which does not involve the next sentence prediction objective, demonstrates competitive performance in ARI tasks (Ruiz-Dolz et al., 2021a), suggesting the absence of this objective does not hinder the model's ability to capture argument relations. Similarly, the ED-DD architectures is relevant to ARI since it allows learning the alignment of pair of sequences (the pair of propositions in ARI). Our result shows that the architecture attains competitive performance only in the sequence-to-sequence alignment task setup. This can be evidenced by the performance improvement of T5 on sequence-to-sequence alignment task over both sequence-classification and token-classification tasks.

These findings highlight the critical significance of tailoring task setups, architectures, and evaluation methodologies to suit the unique intricacies of ARI tasks.

5 Discussion

To contextualise the results reported in the ARIES benchmark, providing a better understanding of their impact in the argument mining community, we compared the best model architecture observed in ARIES with the best performing and most recent identified previous work addressing ARI in each of the datasets individually. Works by Morio et al.

(2022), Ruosch et al. (2022), Chakrabarty et al. (2019), Ruiz-Dolz et al. (2021a), and Kikteva et al. (2023) represent the best possible reference with which to compare ARIES, given the similarity in the way that the ARI task is approached. The resulting comparison is depicted in Table 3.

As it can be observed, ARIES represents a significant jump in performance compared to previous works. Our benchmark consistently outperforms the previously reported results in the most similar instances of the ARI task considering the same eight selected datasets. The direct comparison, however, is difficult to do due to the high variability in which different authors address the task and interpret argumentative concepts. For example, Morio et al. (2022) does it with an end-to-end model, and although we selected the reported results assuming an oracle system for ADU segmentation, the proposed models are not entirely focused on ARI, considering other aspects of argument mining such as component classification. Other works such as (Ruiz-Dolz et al., 2021a) and (Kikteva et al., 2023) consider an additional relation for ARI, the rephrase between two argument propositions. Thus making the ARI a four-class classification problem instead of considering the three classes included in ARIES. Therefore, this comparison needs to be understood as a motivation and a starting point towards a more consistent and unified way of evaluating argument mining systems rather than a direct comparison between works. While worse results in a simpler version of the task should be taken as concerning, worse results in a more complex version of it do not need to mean that the system is worse. With our benchmark, we expect that future contributions in argument mining can be better contextualised and evaluated, moving forward together as a community rather than reporting specific results for heterogeneous setups that are difficult to compare and understand from a broader viewpoint.

Furthermore, we clearly observed how in the CD evaluation of the different natural language modelling approaches and architectures, the performance consistently dropped to the point of being close to the majority baseline. Thus limiting the usability of the resulting models in different domains than the ones included during training. Although some work has investigated cross-domain and cross-language argument mining (Al Khatib et al., 2016; Eger et al., 2018), this issue has never been systematically explored in-depth, leaving the door open to a new challenging direction: robustness in argument mining (Ruiz-Dolz et al., 2024). Considering the relevance of language and domain in natural language argumentation, developing robust systems is a main issue if we want to be able to effectively deploy them in real-world scenarios. For this purpose, ARIES represents a valuable resource, allowing not only to compare between different datasets, but also to measure the cross-dataset robustness of the developed argument mining systems.

Finally, we would also like to mention that recently, Gorur et al. (2024) conducted a thorough study comparing the performance of generative LLMs (i.e., decoder-based architectures) for ARC. Although some of the reported results might seem higher than the ones included in the ARIES benchmark, as noted in the beginning of this paper, relation classification assumes that the relation has already been identified and classifies it as an attack or a support, significantly simplifying the task. Therefore, we excluded these results from our comparison, being a significantly different task highly dependant on a previous step. Instead, ARI represents a completely independent task embedding the main purposes of argument mining (i.e., identifying argument structures from unstructured natural language inputs).

6 Conclusion

In this paper we presented ARIES, a global benchmark for the identification of natural language arguments. ARIES represents an effort to ease the understanding of argument mining contributions and their impact to the community. We achieve this by providing solid results comparing the three main modelling approaches in NLP (i.e., sequence and token classification, and sequence-to-sequence alignment) combined with the three main model architectures (i.e., encoder, decoder, and encoder-decoder). Our benchmark goes all over eight different corpora, presenting new state-of-the-art results for the ARI task, and setting a new reference for research in argument mining. Furthermore, we pointed out the limitations of domain-specific argument mining systems, showing poor results in cross-dataset evaluation. This limitation raises the question of how useful argument extraction systems can be when deployed in the wild, given their limited generalisability, highlighting the need to investigate the robustness of argument mining systems.

	MTC		AAEC		CDCP		ACSP		AMP		AbstRCT		US2016		QT30	
	(1)	ARIES	(1)	ARIES	(1)	ARIES	(2)	ARIES	(3)	ARIES	(1)	ARIES	(4)	ARIES	(5)	ARIES
F1-score	47	**71**	56	**78**	21	**72**	32	**84**	40	**84**	51	**84**	70*	**79**	56*	**85**

Table 3: Comparison of the ARIES benchmark with the previous reported results for ARI in terms of macro-averaged F1-scores. We use * to indicate that the ARI results included rephrase as an additional relation type. For readability purposes we have represented the references in the table as follows (1): (Morio et al., 2022), (2): (Ruosch et al., 2022), (3): (Chakrabarty et al., 2019), (4): (Ruiz-Dolz et al., 2021a), (5): (Kikteva et al., 2023).

As future work, we foresee expanding the ARIES benchmark to more languages than English. Although argument mining has been mostly researched in English, corpora in Catalan (Ruiz-Dolz et al., 2021b), Spanish (Cantador et al., 2020), Japanese (Kimura et al., 2022), or Chinese (Wu et al., 2023) among others have been annotated and publicly released in the recent years. Increasing the language richness in argument mining research can be beneficial, not only for implementing more robust models, but also to help us investigating the differences between relevant natural language argument features underlying different languages.

Acknowledgements

This research is supported in part by: the Swiss National Science Foundation under grant 10001FM_200857; the Volkswagen Foundation under grant 98 543; and the Office of the Director of National Intelligence (ODNI), Intelligence Advanced Research Projects Activity (IARPA), via Contract 2022-22072200004. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of ODNI, IARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

References

Khalid Al Khatib, Henning Wachsmuth, Matthias Hagen, Jonas Köhler, and Benno Stein. 2016. Cross-domain mining of argumentative text through distant supervision. In *Proceedings of the 2016 conference of the north american chapter of the association for computational linguistics: human language technologies*, pages 1395–1404.

Jianzhu Bao, Yuhang He, Yang Sun, Bin Liang, Jiachen Du, Bing Qin, Min Yang, and Ruifeng Xu. 2022. A generative model for end-to-end argument mining with reconstructed positional encoding and constrained pointer mechanism. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10437–10449.

Francesco Barbieri, Jose Camacho-Collados, Luis Espinosa Anke, and Leonardo Neves. 2020. Tweeteval: Unified benchmark and comparative evaluation for tweet classification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1644–1650.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Elena Cabrio and Serena Villata. 2014. Node: A benchmark of natural language arguments. In *Computational Models of Argument*, pages 449–450. IOS Press.

Iván Cantador, María E Cortés-Cediel, and Miriam Fernández. 2020. Exploiting open data to analyze discussion and controversy in online citizen participation. *Information Processing & Management*, 57(5):102301.

Tuhin Chakrabarty, Christopher Hidey, Smaranda Muresan, Kathleen Mckeown, and Alyssa Hwang. 2019. Ampersand: Argument mining for persuasive online discussions. 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 2933–2943.

Oana Cocarascu, Elena Cabrio, Serena Villata, and Francesca Toni. 2020. Dataset independent baselines for relation prediction in argument mining. In *COMMA 2020-8th International Conference on Computational Models of Argument*, volume 326, pages 45–52.

Oana Cocarascu and Francesca Toni. 2017. Identifying attack and support argumentative relations using deep learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1374–1379.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for*

Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186.

Steffen Eger, Johannes Daxenberger, and Iryna Gurevych. 2017. Neural end-to-end learning for computational argumentation mining. *arXiv preprint arXiv:1704.06104*.

Steffen Eger, Johannes Daxenberger, Christian Stab, and Iryna Gurevych. 2018. Cross-lingual argumentation mining: Machine translation (and a bit of projection) is all you need! In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 831–844.

Deniz Gorur, Antonio Rago, and Francesca Toni. 2024. Can large language models perform relation-based argument mining? *arXiv preprint arXiv:2402.11243*.

Annette Hautli-Janisz, Zlata Kikteva, Wassiliki Siskou, Kamila Gorska, Ray Becker, and Chris Reed. 2022. Qt30: A corpus of argument and conflict in broadcast debate. In *Proceedings of the 13th Language Resources and Evaluation Conference*, pages 3291–3300. European Language Resources Association (ELRA).

Masayuki Kawarada, Tsutomu Hirao, Wataru Uchida, and Masaaki Nagata. 2024. Argument mining as a text-to-text generation task. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2002–2014.

Zlata Kikteva, Alexander Trautsch, Patrick Katzer, Mirko Oest, Steffen Herbold, and Annette Hautli. 2023. On the impact of reconstruction and context for argument prediction in natural debate. In *Proceedings of the 10th Workshop on Argument Mining*, pages 100–106.

Yasutomo Kimura, Hokuto Ototake, and Minoru Sasaki. 2022. Budget argument mining dataset using japanese minutes from the national diet and local assemblies. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 6131–6138.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Marcin Koszowy, Steve Oswald, Katarzyna Budzynska, Barbara Konat, and Pascal Gygax. 2022. A pragmatic account of rephrase in argumentation: linguistic and cognitive evidence. *Informal Logic*, 42(1):49–82.

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

John Lawrence and Chris Reed. 2020. Argument mining: A survey. *Computational Linguistics*, 45(4):765–818.

Marco Lippi and Paolo Torroni. 2016. Argumentation mining: State of the art and emerging trends. *ACM Transactions on Internet Technology (TOIT)*, 16(2):1–25.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Tobias Mayer, Elena Cabrio, and Serena Villata. 2020. Transformer-based argument mining for healthcare applications. In *ECAI 2020*, pages 2108–2115. IOS Press.

Raquel Mochales and Marie-Francine Moens. 2011. Argumentation mining. *Artificial intelligence and law*, 19:1–22.

Gaku Morio, Hiroaki Ozaki, Terufumi Morishita, and Kohsuke Yanai. 2022. End-to-end argument mining with cross-corpora multi-task learning. *Transactions of the Association for Computational Linguistics*, 10:639–658.

Joonsuk Park and Claire Cardie. 2018. A corpus of erulemaking user comments for measuring evaluability of arguments. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Andreas Peldszus and Manfred Stede. 2015. An annotated corpus of argumentative microtexts. In *Argumentation and Reasoned Action: Proceedings of the 1st European Conference on Argumentation, Lisbon*, volume 2, pages 801–815.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Ramon Ruiz-Dolz, Jose Alemany, Stella M Heras Barberá, and Ana García-Fornes. 2021a. Transformer-based models for automatic identification of argument relations: A cross-domain evaluation. *IEEE Intelligent Systems*, 36(6):62–70.

Ramon Ruiz-Dolz, Chr-Jr Chiu, Chung-Chi Chen, Noriko Kando, and Hsin-Hsi Chen. 2024. Learning strategies for robust argument mining: An analysis of variations in language and domain. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 10286–10292.

Ramon Ruiz-Dolz, Montserrat Nofre, Mariona Taulé, Stella Heras, and Ana García-Fornes. 2021b. Vivesdebate: A new annotated multilingual corpus of argumentation in a debate tournament. *Applied Sciences*, 11(15):7160.

Florian Ruosch, Cristina Sarasua, and Abraham Bernstein. 2022. Bam: Benchmarking argument mining on scientific documents. CEUR Workshop Proceedings.

Florian Ruosch, Cristina Sarasua, and Abraham Bernstein. 2023. Dream: Deployment of recombination and ensembles in argument mining. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 5277–5290. Association for Computational Linguistics.

Lida Shi, Fausto Giunchiglia, Rui Song, Daqian Shi, Tongtong Liu, Xiaolei Diao, and Hao Xu. 2022. A simple contrastive learning framework for interactive argument pair identification via argument-context extraction. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10027–10039.

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

Felix Stahlberg. 2020. Neural machine translation: A review. *Journal of Artificial Intelligence Research*, 69:343–418.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Jacky Visser, Barbara Konat, Rory Duthie, Marcin Koszowy, Katarzyna Budzynska, and Chris Reed. 2020. Argumentation in the 2016 us presidential elections: annotated corpora of television debates and social media reaction. *Language Resources and Evaluation*, 54(1):123–154.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355.

Hongyi Wu, Xinshu Shen, Man Lan, Shaoguang Mao, Xiaopeng Bai, and Yuanbin Wu. 2023. A multi-task dataset for assessing discourse coherence in chinese essays: Structure, theme, and logic analysis. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Shu Wen Yang, Po Han Chi, Yung Sung Chuang, Cheng I Jeff Lai, Kushal Lakhotia, Yist Y Lin, Andy T Liu, Jiatong Shi, Xuankai Chang, Guan Ting Lin, et al. 2021. Superb: Speech processing universal performance benchmark. In *22nd Annual Conference of the International Speech Communication Association, INTERSPEECH 2021*, pages 3161–3165. International Speech Communication Association.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and William B Dolan. 2020. Dialogpt: Large-scale generative pre-training for conversational response generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 270–278.

A Experiment Setup

A.1 Training Procedure

Hyper-parameters: We employ Adam optimisation (Kingma and Ba, 2014) to minimise the cost function, using a learning rate of 2×10^{-5} and categorical cross-entropy loss and a batch size of 16.

Gradient Clipping: To prevent exploding gradients during training, we applied gradient clipping. We used a maximum gradient norm (`max_grad_norm`) parameter to determine the threshold for gradient clipping.

Warm-up and Learning Rate Schedule: We employed a linear warm-up strategy for the learning rate. The number of warm-up steps was set to 10% of the total training steps. Following the warm-up phase, the learning rate schedule was determined by a lambda function. This function linearly increases the learning rate during the warm-up phase and decreases it linearly thereafter.

A.1.1 Input Setup

For the sequence classification task, we combine the premise and conclusion using a special token `[SEP]`. In the sequence-alignment task, the encoder receives the premise while the decoder processes the conclusion separately. The token-classification task is provided with the entire argument along with one of the propositions (say the premise). To ensure consistency across architectures, the maximum input length is set to 512. In the sequence-to-sequence alignment task, where inputs are provided separately to the encoder and decoder, we set the maximum input size to 256 for both components to enable direct comparison. For the token-classification configuration, if the input length exceeds 512, we extract a span of the argument relevant to the premise and conclusion. Initially, we measure the size of one proposition (the premise) and if the combined size of the argument and premise is less than 512, we use the entirety of both. Otherwise, we use the following heuristic to extract the relevant part of the argument: extract a span of argument involving both the premise and conclusion if the size of the span and the premise is less than 512. If not, expand the span in the direction of the conclusion until the size constraint is met and append the conclusion to the argument span.

A.2 Model Configurations

To facilitate direct comparisons between architectures and configurations, we ensure comparable model sizes across all setups. Specifically, we employ RoBERTa-large (Liu et al., 2019) (355 million parameters) for the ED, DialoGPT-medium (Zhang et al., 2020) (345 million parameters) for the DD, and T5-base (Raffel et al., 2020) (220 million parameters) for the ED-DD configuration.

A.3 Sequence-to-sequence Alignment Task

For the sequence-to-sequence alignment task, we try two configurations. First, we leverage the `T5ForConditionalGeneration`[1] implementation, fine-tuned to generate conclusions given premises. We also concatenate the final hidden state of the encoder with that of the decoder which is then fed into a linear layer to predict the argument relation between the premise and conclusion. In an alternative approach, we employ the `T5ForSequenceClassification`[2] implementation, where the model is fine-tuned in the identification of argument relations, without the added complexity of conclusion generation given a premise. Across the configurations, the premise is provided as input to the encoder, while the conclusion serves as the input to the decoder. Our experiment reveals that T5ForConditionalGeneration configuration provides better result and all the experimental results on the SeqAln task is reported based on this configuration.

A.4 Sequence Classification Task

For the ED architecture, we utilise the final output of the HuggingFace implementation of RoBERTaForSequenceClassification[3]. Similarly, DD architecture, we leverage the final output of the HuggingFace implementation of DialoGPTForSequenceClassification [4]. For the ED-DD, we use the final output of the decoder based on the HuggingFace implementation of T5ForSequenceClassification[5]. Across the sequence classification task, the input to the respective models is the concatenation of the premise and conclusion.

[1] https://huggingface.co/docs/transformers/en/model_doc/t5#transformers.T5ForConditionalGeneration

[2] https://huggingface.co/docs/transformers/en/model_doc/t5#transformers.T5ForSequenceClassification

[3] https://huggingface.co/docs/transformers/en/model_doc/RoBERTa#transformers.RoBERTaForSequenceClassification

[4] https://huggingface.co/docs/transformers/en/model_doc/dialogpt

[5] https://huggingface.co/docs/transformers/en/model_doc/t5#transformers.T5ForSequenceClassification

Model	Train Data	AAEC	ACSP	ABstRACT	US2016	QT30	CDCP	MTC	AMP
DialogPT	AAEC	-	0.402	0.473	0.462	0.410	0.454	0.465	0.573*
	CDCP	0.365	0.390	0.432	0.425	0.390	-	0.312	0.564*
	ACSP	0.413	-	0.425	0.413	0.434	0.336	0.467	0.562*
	QT30	0.470	0.479	0.472	0.479	-	0.480	0.467	0.553*
	ABstRACT	0.281	0.342	-	0.365	0.340	0.400	0.435	0.610*
	MTC	0.363	0.291	0.434	0.356	0.316	0.381	-	0.631*
	US2016	0.461	0.430	0.424	-	0.463	0.471	0.461	0.563*
	AMP	0.532*	0.551*	0.523*	0.574*	0.621*	0.465*	0.346*	-
RoBERTa	AAEC	-	0.390	0.459	0.399	0.446	0.454	0.535	0.561*
	CDCP	0.322	0.312	0.411	0.403	0.373	-	0.379	0.562*
	ACSP	0.479	-	0.489	0.520	0.560	0.379	0.504*	0.542*
	QT30	0.388	0.370	0.491	0.501	-	0.405	0.479	0.523*
	ABstRACT	0.332	0.358	-	0.345	0.362	0.475	0.491	0.586*
	MTC	0.309	0.302	0.319	0.361	0.331	0.284	-	0.542*
	US2016	0.399	0.426	0.512	-	0.456	0.420	0.420	0.571*
	AMP	0.512*	0.551*	0.502*	0.566*	0.614*	0.479*	0.348*	-
T5	AAEC	-	0.306	0.342	0.395	0.339	0.355	0.390	0.491*
	CDCP	0.356	0.362	0.363	0.355	0.368	-	0.261	0.501*
	ACSP	0.304	-	0.378	0.336	0.339	0.305	0.444	0.456*
	QT30	0.351	0.322	0.344	0.359	-	0.349	0.419	0.541*
	ABstRACT	0.342	0.305	-	0.320	0.333	0.376	0.376	0.511*
	MTC	0.319	0.312	0.346	0.351	0.359	0.315	-	0.529*
	US2016	0.345	0.328	0.364	-	0.389	0.399	0.355	0.473*
	AMP	0.486*	0.462*	0.396*	0.421*	0.441*	0.365*	0.245*	-

Table 4: CD evaluation performance of each model architecture on the SeqCls task setup. We use * to denote that the evaluation results reported on the AMP represent binary predictions.

Model	**Train Data**	**AAEC**	**ACSP**	**ABstRACT**	**US2016**	**QT30**	**CDCP**	**MTC**
`DialogPT`	AAEC	-	0.312	0.40	0.441	0.467	461	0.479
	CDCP	0.277	0.285	0.355	0.376	0.335	-	0.335
	ACSP	0.368	-	0.418	0.427	0.414	0.319	0.366
	QT30	0.346	0.358	0.268	0.500	-	0.479	0.267
	ABstRACT	0.334	0.311	-	0.377	0.320	0.322	0.423
	MTC	0.347	0.297	0.397	0.423	0.322	0.274	-
	US2016	0.389	0.378	0.287	-	0.519	0.400	0.279
`RoBERTa`	AAEC	-	0.294	0.447	0.440	0.433	0.442	0.450
	CDCP	0.267	0.265	0.334	0.341	0.307	-	0.323
	ACSP	0.349	-	0.411	0.411	0.407	0.300	0.337
	QT30	0.328	0.316	0.256	0.509	-	0.237	0.238
	ABstRACT	0.290	0.297	-	0.354	0.319	0.291	0.404
	MTC	0.335	0.264	0.380	0.417	0.336	0.286	-
	US2016	0.311	0.307	0.230	-	0.359	0.246	0.246
`T5`	AAEC	-	0.269	0.365	0.365	0.342	366	0.365
	CDCP	267	0.279	0.352	0.361	0.307	-	0.342
	ACSP	0.332	-	0.411	0.401	0.413	0.281	0.332
	QT30	0.321	0.298	0.241	0.486	-	0.423	0.237
	ABstRACT	0.265	0.282	-	0.361	0.324	0.318	0.413
	MTC	0.323	0.264	0.380	0.421	0.336	0.286	-
	US2016	0.333	0.317	0.251	-	0.522	0.398	0.266

Table 5: CD evaluation performance of each model architecture on the TokCls task setup.