# Parsing Argumentative Structure in English-as-Foreign-Language Essays

**Jan Wira Gotama Putra**[†], **Simone Teufel**[‡†], **Takenobu Tokunaga**[†]
[†]Tokyo Institute of Technology, Japan
[‡]University of Cambridge, United Kingdom
{gotama.w.aa@m, take@c}.titech.ac.jp, simone.teufel@cl.cam.ac.uk

## Abstract

This paper presents a study on parsing the argumentative structure in English-as-foreign-language (EFL) essays, which are inherently noisy. The parsing process consists of two steps, linking related sentences and then labelling their relations. We experiment with several deep learning architectures to address each task independently. In the sentence linking task, a biaffine model performed the best. In the relation labelling task, a fine-tuned BERT model performed the best. Two sentence encoders are employed, and we observed that non-fine-tuning models generally performed better when using Sentence-BERT as opposed to BERT encoder. We trained our models using two types of parallel texts: original noisy EFL essays and those improved by annotators, then evaluate them on the original essays. The experiment shows that an end-to-end in-domain system achieved an accuracy of .341. On the other hand, the cross-domain system achieved 94% performance of the in-domain system. This signals that well-written texts can also be useful to train argument mining system for noisy texts.

## 1 Introduction

Real-world texts are not always well-written, especially in the education area where students are still learning how to write effectively. It has been observed that student texts often require improvements at the discourse-level, e.g., in persuasiveness and content organisation aspects (Bamberg, 1983; Zhang and Litman, 2015; Carlile et al., 2018). Worse still, texts written by non-native speakers are also less coherent, exhibit less lexical richness and more unnatural lexical choices and collocations (Johns, 1986; Silva, 1993; Rabinovich et al., 2016). Our long-term goal is to improve EFL essays from the discourse perspective. One way to do this is by recommending a better arrangement of sentences, which enhances text coherence and comprehension (Connor, 2002; Bacha, 2010). This may serve as feedback for students in the educational setting (Invanic, 2004). The first step to achieve this goal, which is discussed in the current paper, is parsing argumentative structure in terms of dependencies between sentences. This is because the relationships between sentences are crucial to determine the proper order of sentences (Grosz and Sidner, 1986; Hovy, 1991; Webber and Joshi, 2012).

This paper describes the application of Argument Mining (AM) to EFL essays. AM is an emerging area in computational linguistics which aims to explain how argumentative discourse units (e.g., sentences, clauses) function and relate to each other in the discourse, forming an argument as a whole (Lippi and Torroni, 2016). AM has broad applications in various areas, such as in the legal (Ashley, 1990) and news (Al-Khatib et al., 2016) domains. Also in the education domain, AM is beneficial for many downstream tasks such as text assessment (Wachsmuth et al., 2016), text improvement (as described above) and teaching (Putra et al., 2020). It is common in AM to use well-written texts written by proficient authors, as do Peldszus and Stede (2016), Al-Khatib et al. (2016), among others. However, there are more non-native speakers of English than native speakers in the world, and their writings are often noisy as previously described. Yet, EFL is a niche domain in AM.

This paper presents three contributions. First, this paper presents an application of AM to EFL essays. We parse the argumentative structure in two steps: (i) a ***sentence linking*** step where we identify related sentences that should be linked, forming a tree structure, and (ii) a ***relation labelling*** step, where we label the relationship between the sentences. Several deep learning models were evaluated to address each step. We do not only evaluate the model performance based on individual links



*Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 97–109
April 20, 2021 

but also perform a structural analysis, giving more insights into the models' ability to learn different aspects of the argumentative structure.

The second contribution is showing the effectiveness of well-written texts as training data for argumentative parsing of noisy texts. Many AM corpora exist for well-written texts because past studies typically assumed well-written input. Corpora with noisy texts, such as the EFL one we use here, exist, but are far more infrequent. In the past, well-written and noisy texts have been treated as two separate domains, and AM systems were trained separately on each domain. We want to investigate how far the existing labelled corpora for well-written texts can also be useful for training parsers for noisy texts. To this end, we train parsers on both in-domain and out-of-domain texts and evaluate them on the in-domain task. For our out-of-domain texts, we use the improved versions of noisy EFL texts. These improvements were produced by an expert annotator and have a quality closer to those of proficient authors.

The third contribution of this paper is an evaluation of Sentence-BERT (SBERT, Reimers and Gurevych (2019)) in AM as a downstream application setting. BERT (Devlin et al., 2019) is a popular transformer-based language model (LM), but as it is designed to be fine-tuned, it can be suboptimal in low-resource settings. SBERT tries to alleviate this problem by producing a more universal sentence embeddings, that can be used as they are in many tasks. The idea of training embeddings on the natural language inference (NLI) task goes back to Conneau et al. (2017), and this is the SBERT variant we use here. The NLI task involves recognising textual entailment (TE), and a TE model has been previously used by Cabrio and Villata (2012) for argumentation. We will quantify how the two encoders perform in our task. All resources of this paper are available on github.[1]

## 2 Related Work

Argumentative structure analysis consists of two main steps (Lippi and Torroni, 2016). The first step is *argumentative component identification* (ACI), which segments a text into argumentative discourse units (ADUs); then differentiates them into argumentative (ACs) and non-argumentative components (non-ACs). ACs function argumentatively while non-ACs do not, e.g., describing a personal episode in response to the given writing prompt. ACs can be further classified according to their communicative roles, e.g., *claim* and *premise*. The second step is *argumentative structure prediction*, which contains two subtasks: (1) *linking* and (2) *relation labelling*. In the linking task, directed relations are established from *source* to *target* ACs to form a structured representation of the text, often in the form of a tree. In the relation labelling task, we identify the relations that connect them, e.g., *support* and *attack*.

In the education domain, argumentative structure interrelates with text quality, and it becomes one of the features that go into automatic essay scoring (AES) systems (Persing et al., 2010; Song et al., 2014; Ghosh et al., 2016; Wachsmuth et al., 2016). End-to-end AES systems also exist, but hybrid models are preferred for both performance and explainability reasons (Uto et al., 2020).

Eger et al. (2017) formulated AM in three ways: as relation extraction, as sequence tagging and as dependency parsing. They performed end-to-end AM at token-level, executing all subtasks in AM all at once. Eger et al. achieved the highest performance in their experiments with the relation extraction model LSTM-ER (Miwa and Bansal, 2016). We instead use their sequence tagging formulation, which adapts the existing vanilla **B**idirectional **L**ong-**s**hort-**t**erm **m**emory (BiLSTM) network (Hochreiter and Schmidhuber, 1997; Huang et al., 2015), as it can be straightforwardly applied to our task. The dependency parsing formulation is also a straightforward adaptation as it models tree structures. The biaffine model is the current state-of-the-art of syntactic dependency parsing (Dozat and Manning, 2017), and it has been adapted to relation detection and labelling tasks in AM by Morio et al. (2020). In a similar way, we also adapt the biaffine model to our argumentative structure. However, we use sentences instead of spans as ADU, trees instead of graphs.

Most work in AM uses well-written texts in the legal (e.g., Ashley, 1990; Yamada et al., 2019) and news (e.g., Al-Khatib et al., 2016) domains, but there are several AM studies that concentrate on noisy texts. For example, Habernal and Gurevych (2017) focused on the ACI task in web-discourse. Morio and Fujita (2018) investigated how to link arguments in discussion threads. In the education domain, Stab and Gurevych (2017) studied the argumentation in persuasive essays. One of the prob-

[1] https://github.com/wiragotama/BEA2021

lems with the existing corpora is the unclear distinction between native and non-native speakers. Additionally, to investigate and bridge the gap of performance between AM systems on noisy and well-written texts, it is necessary to use a parallel corpus containing both versions of texts. However, none of the above studies did.

## 3 Dataset

We use part of the "**I**nternational **C**orpus **N**etwork of **A**sian **L**earners of **E**nglish" (Ishikawa, 2013, 2018), which we annotated with **A**rgumentative **S**tructure and **S**entence **R**eordering ("ICNALE-AS2R" corpus).[2] This corpus contains 434 essays written by college students in various Asian countries. They are written in response to two prompts: (1) about banning smoking and (2) about students' part-time jobs. Essays are scored in the range of $[0, 100]$. There are two novelties in this corpus: (1) it uses a new annotation scheme as described below and (2) contains a parallel version of essays which have been improved from the discourse perspective. Therefore, this corpus can be used in many downstream tasks, e.g., employing argumentative structures for assessing and improving EFL texts. It is also possible to extend the improved version of texts on other linguistic aspects.

The corpus was annotated at sentence-level, i.e., a sentence corresponds to an ADU.[3] In our annotation scheme, we first differentiate sentences as ACs and non-ACs, without further classification of AC roles. Annotators then establish links from source to target ACs, forming tree-structured representations of the texts. Then, they identify the relations that connect ACs. We instructed annotators to use the *major claim*, the statement that expresses the essay author's opinion at the highest level of abstraction, as the *root* of the structure. As there are no further classification of AC roles, the term "major claim" here refers to a concept, not an explicitly annotated category. As the last step, annotators rearrange sentences and performed *text repair* to improve the texts from a discourse perspective.

There are four relations between ACs: SUPPORT (sup), ATTACK (att), DETAIL (det) and RESTATEMENT (res). SUPPORT and ATTACK relations are common in AM. They are used when the source sentence supports or attacks the argument in the target sentence (Peldszus and Stede, 2013; Stab and Gurevych, 2014). We use the DETAIL relation in two cases. First, when the source presents additional details (further explanations, descriptions or elaborations) about the target sentence, and second, when the source introduces a topic of the discussion in a neutral way by providing general background. From the organisational perspective, the differentiation between DETAIL and SUPPORT is useful. While the source sentence in a SUPPORT relation *ideally* follows its target, the DETAIL relation has more flexibility. We also use a relation called RESTATEMENT for those situations where high-level parts of an argument are repeated or summarised for the second time, e.g., when the major claim is restated in the conclusion of the essay. DETAIL and RESTATEMENT links are not common in AM; the first was introduced by Kirschner et al. (2015) and the second by Skeppstedt et al. (2018), but both work on well-written texts. The combination of these four relations is unique in AM.

To improve the texts, annotators were asked to rearrange sentences so that it results in the most logically well-structured texts they can think of. This is the second annotation layer in our corpus. No particular reordering strategy was instructed. Reordering, however, may cause irrelevant or incorrect referring and connective expressions (Iida and Tokunaga, 2014). To correct these expressions, annotators were instructed to minimally *repair* the text where this is necessary to retain the original meaning of the sentence. For instance, they replaced pronouns with their referents, and removed or replaced inappropriate connectives. Text repair is also necessary to achieve standalone major claims. For example, "*I think **so***" with *so* referring to the writing prompt (underlined in what follows) can be rephrased as "*I think smoking should be banned at all restaurants*."

Figure 1 shows an example of our annotation scheme using a real EFL essay. Figure 2 then illustrates how the reordering operation produced an improved essay. The annotator recognised that (16) is the proper major claim of the essay in Figure 1. However, this essay is problematic because the major claim is not introduced at the beginning of the essay. Thus, the annotator moved (16) to the beginning, and the whole essay is concluded by sentence

[2] A full description of the corpus and the annotation study we performed is available in a separate submission.

[3] Texts written by proficient authors may contain two or more ideas per sentence. However, our targets are EFL learners; pedagogically, they are often taught to put one idea per sentence.

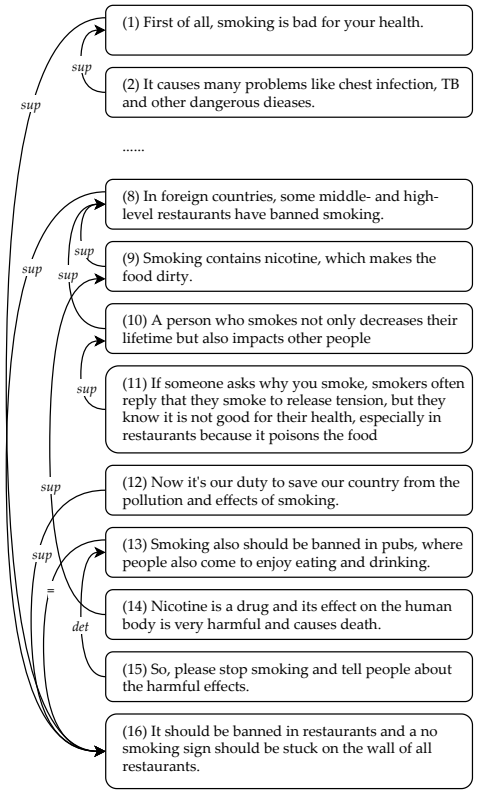



Figure 1: A snippet of argumentative structure annotation for essay code "W_PAK_SMK0_022_B1_1_EDIT" by our expert annotator. The essay discusses banning smoking in restaurants.

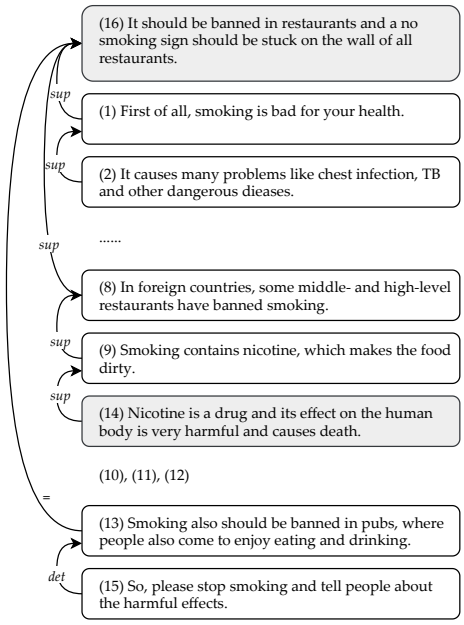



Figure 2: The improved version of essay Figure 1.

(13) in Figure 2. We also observe that crossing links happen in Figure 1, and they may suggest the jump of ideas, indicating coherence breaks. For example, sentence (14) describes nicotine and the annotator thinks that it properly connects to (9) which also talks about nicotine. Therefore, it is more desirable to place sentences (9) and (14) close to each other, as shown in Figure 2. The reordered version is arguably better since it is more consistent with the argumentative-development-strategy prescribed in teaching, i.e., introduce a topic and state the author's stance on that topic, support the stance by presenting more detailed reasons, and finally concludes the essay at the end (Silva, 1993; Bacha, 2010).

We performed an inter-annotator agreement (IAA) study on 20 essays, using as annotators a PhD student in English education (also an EFL teacher – *expert annotator*) and the first author, both having a near-native English proficiency. We found the agreement to be Cohen's $\kappa$=.66 for ACI; $\kappa$=.53 for sentence linking; and $\kappa$=.61 for relation labelling (Cohen, 1960). The sentence linking task was the most difficult one, and it is understandable since a text may have multiple acceptable structures. The relation labels hardest to distinguish were between SUPPORT and DETAIL.

This kind of annotation is expensive. Also, there is no metric to measure the agreement on reordering. Therefore, we chose to have the expert annotator annotate all essays in the production for the purpose of consistency. There are 6,021 sentences in the ICNALE-AS2R corpus; 5,799 (96%) of these are ACs and 222 (4%) are non-ACs. An essay contains 14 sentences on average, and the average structural depth is 4.3 (counting the root at depth $= 0$). The corpus does not contain paragraph breaks. The most frequent relation label is SUPPORT (3,029–56%), followed by DETAIL (1,585–30%), ATTACK (437–8%), and RESTATEMENT (314–6%). In total, 105 out of 434 essays were rearranged (1-3 sentences were moved on average). As we have explained before, the expert annotator reordered a scattered set of sentences which logically form a sub-argument to be close in position to each other. They also placed the major claim at the beginning as opposed to the middle or the end of the essay. In general, the expert arranges the essays to be more consistent with the typical argumentative-development strategy prescribed in teaching. The text repair was done on 181 sentences, 123 (71%) of which attempt to repair the prompt-type error of the major claim. The remain-

ing 58 sentences concern changes in connectives and referring expressions.

## 4 Parsing Models

We adopt a pipeline approach by using independent models for sentence linking, which includes the ACI task, and relation labelling. Although a pipeline system may fall prey to error propagation, for a new scheme and corpus, it can be advantageous to look at intermediate results.

### 4.1 Sentence Linking

Given an entire essay as a sequence of sentences $s_1, ..., s_N$, our sentence linking model outputs the distance $d_1, ..., d_N$ between each sentence $s_i$ to its target; if a sentence is connected to its preceding sentence, the distance is $d = -1$. We consider those sentences that have no explicitly annotated outgoing links as linked to themselves ($d = 0$); this concerns major claims (roots) and non-ACs.

| $\leq -5$ | $-4$ | $-3$ | $-2$ | $-1$ | 0 |
|---|---|---|---|---|---|
| 16.6 | 3.9 | 5.2 | 8.3 | 37.0 | 10.9 |
| $\geq +5$ | $+4$ | $+3$ | $+2$ | $+1$ | |
| 1.0 | 0.6 | 0.9 | 2.3 | 13.4 | |

Table 1: Distribution of distance (in percent) between source and target sentences in the corpus.

Table 1 shows the distribution of distance between the source and target sentences in the corpus, which ranges $[-26, ..., +15]$. Adjacent links predominate ($50.4\%$). Short-distance links ($2 \leq |d| \leq 4$) make up $21.2\%$ of the total. Backward long distance links at $d \leq -5$ are $16.6\%$, whereas forward long distance links are rare ($1.0\%$).

We follow the formulation by Eger et al. (2017), where AM is modelled as sequence tagging (4.1.1) and as dependency parsing (4.1.2).

#### 4.1.1 Sequence Tagger Model

Figure 3 shows our sequence tagging architecture ("SEQTG"). We adapt the vanilla BiLSTM with softmax prediction layers (as Eger et al. (2017) similarly did), training the model in a multi-task learning (MTL) setup. There are two prediction layers: (1) for sentence linking as main task and (2) for ACI as an auxiliary task.

The input sentences $s_1, ..., s_N$ are first encoded into their respective sentence embeddings (using either BERT or SBERT as encoder).[4] We do not fine-tune the encoder because our dataset is too small for it.[5] The resulting sentence embeddings are then fed into a dense layer for dimensionality reduction. The results are fed into a BiLSTM layer ($\#stack = 3$) to produce contextual sentence representations, then fed into prediction layers.

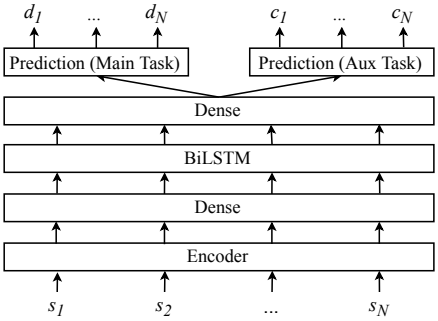



Figure 3: BiLSTM-softmax ("SEQTG").

**Main Task:** The model predicts the probability of link distances, in the range $[-26, ..., +15]$. To make sure there is no out-of-bound prediction, we perform a constrained argmax during prediction time. For each sentence $s_i$, we compute the argmax only for distances at $[1 - i, ..., N - i]; i \geq 1$.

**Auxiliary Task:** As the auxiliary task, the model predicts *quasi*-argumentative-component type $c$ for each input sentence. Our scheme does not assign AC roles per se, but we can compile the following sentence types from the tree typology:

- *major claim (root)*: only incoming links,
- *AC (non-leaf)*: both outgoing and incoming links,
- *AC (leaf)*: only outgoing links, and
- *non-AC*: neither incoming nor outgoing links.

These four labels should make a good auxiliary task as they should help the model to learn the placement of sentences in the hierarchical structure.

We use the dynamic combination of loss as the MTL objective (Kendall et al., 2018). To evaluate whether the auxiliary task does improve the model performance, we also experiment only on the main task (single-task learning–STL).

#### 4.1.2 Biaffine Model

We adapt the biaffine model ("BIAF") by Dozat and Manning (2017), treating the sentence linking task as sentence-level dependency parsing (Figure 4).

The first three layers produce contextual sentence representations in the same manner as in the

[4] By averaging subword embeddings.

[5] We conducted a preliminary fine-tuning experiment on sentence linking task, but the performance did not improve.

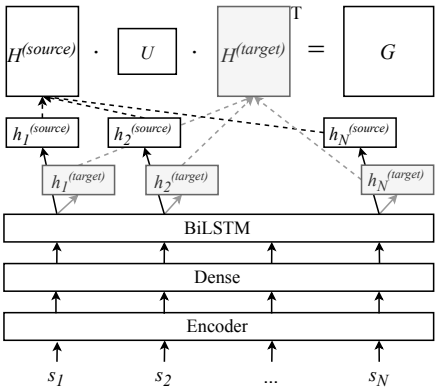



Figure 4: Biaffine Model ("BIAF").

SEQTG model. These representations are then fed into two different dense layers, in order to encode the corresponding sentence when it acts as a source $\left(h^{(source)}\right)$ or target $\left(h^{(target)}\right)$ in a relation. Finally, a biaffine transformation is applied to all source and target representations to produce the final output matrix $G \in \mathbb{R}^{N \times N}$, in which each row $g_i$ represents where the source sentence $s_i$ should point to (its highest scoring target).

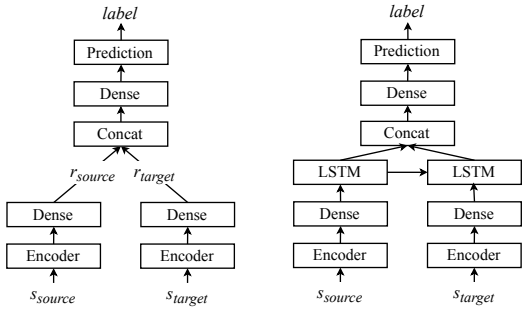
When only considering the highest scoring or most probable target for each source sentence in isolation, the output of the models (SEQTG and BIAF) does not always form a tree (30-40% non-tree outputs in our experiment). In these cases, we use Chu-Liu-Edmonds algorithm (Chu and Liu, 1965; Edmonds, 1967) to create a minimum spanning tree out of the output.

### 4.2 Relation Labelling

In the relation labelling task, given a pair of *linked* source and target sentences $\langle s_{source}, s_{target} \rangle$, a model outputs the label that connects them, i.e., one of {SUPPORT, ATTACK, DETAIL, RESTATEMENT}. We use non-fine-tuning models with feed-forward architecture and fine-tuning transformer-based LMs.

#### 4.2.1 Non-fine-tuning Models

In non-fine-tuning models, both source and target sentences $\langle s_{source}, s_{target} \rangle$ are encoded using BERT or SBERT to produce their respective embeddings. We then pass these embeddings into respective dense layers for a dimensionality reduction and transformation step, producing $\langle r_{source}, r_{target} \rangle$. As the first option ("FFCON", Figure 5a), $r_{source}$ and $r_{target}$ are concatenated, passed to a dense layer for a further transformation, and finally fed into a prediction layer. As the second option (FFLSTM, Figure 5b), we feed $r_{source}$ and $r_{target}$ to an LSTM layer, and the hidden units of LSTM are concatenated before being sent to a dense layer (Deguchi and Yamaguchi, 2019).




(a) FFCON model. (b) FFLSTM model.

Figure 5: Non-finetuning relation labelling models.

#### 4.2.2 Fine-tuning Models

Unlike sentence linking, where an entire essay is taken as input, the relation labelling task takes a pair of sentences. There are 5,365 of such pairs in the ICNALE-AS2R corpus. We fine-tune BERT and DISTILBERT (Sanh et al., 2019) on the resulting sentence pair classification task. The pair is fed into the transformer model, and then the `[CLS]` token representation is passed into a prediction layer.

## 5 Experimental Results and Discussion

The dataset is split into 80% training set (347 essays , 4,841 sentences) and 20% testing set (87 essays, 1,180 sentences), stratified according to prompts, scores and country of origin of the EFL learners. We are interested in how the AM models trained on well-written texts may perform on more noisy texts. To find out, we train the models on both the original EFL texts (in-domain) and the parallel improved texts (out-of-domain), then evaluated on the original EFL texts. The difference between in- and out-of-domain data lies on the textual surface, i.e., sentence rearrangement, the use of connectives, referring expressions, and textual repair for major claims. Since not all essays undergo any reordering, the out-of-domain data is roughly 75% the same as the in-domain data.

The number of hidden units and learning rates (alongside other implementation notes) to train our models can be found in Appendix A. We run the experiment for 20 times,[6] and report the average

[6]Using the same dataset split. This is to account for ran-

performance. The relation labelling models are trained and evaluated using sentence pairs according to the gold-standard. In the end-to-end evaluation (Section 5.3), however, the input to the relation labelling model is the automatic prediction. Statistical testing, whenever possible, is conducted using the student's t-test (Fisher, 1937) on the performance scores of the 20 runs, with a significance level of $\alpha = 0.05$.

### 5.1 Sentence Linking

We first report our in-domain before turning to the cross-domain results.

Table 2 shows our experimental result on the prediction of individual links. The best model is a biaffine model, namely SBERT-BIAF, statistically outperforming the next-best non-biaffine model (accuracy .471 vs .444 and F1-macro .323 vs .274; significant difference on both metrics). Training the SEQTG model in the MTL setting did not improve the performance on these standard metrics.

| Model | Accuracy | F1-macro |
|---|---|---|
| BERT-SEQTG [STL] | .436 | .274 |
| BERT-SEQTG [MTL] | .431 | .242 |
| BERT-BIAF | .446 | .310 |
| SBERT-SEQTG [STL] | .444 | .229 |
| SBERT-SEQTG [MTL] | .438 | .220 |
| SBERT-BIAF | **.471**$^{\dagger}$ | **.323**$^{\dagger}$ |

Table 2: In-domain results of individual-link predictions in the sentence linking task. Best result shown in **bold-face**. The † symbol indicates that the difference to the second-best result (underlined) is significant.

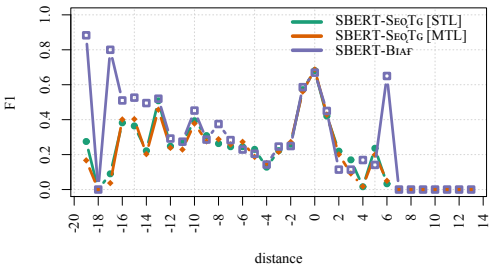



Figure 6: Models' performance across distances for in-domain evaluation using SBERT encoder.

To gain deeper insights into model quality, we also considered the models' F1 score per target distance (Figure 6). All models, and in particular BIAF, are better at predicting long-distance links ($d \leq -5$, avg. F1 = [0.22, 0.41]) than short distance links ($2 \leq |d| \leq 4$, avg. F1 = [0.16, 0.24]) when using SBERT encoder (the same trend goes when using BERT encoder). Long-distance links tend to happen at the higher tree level, e.g., the links from nodes at depth=1 to the root, while short-distance links tend to happen at the deeper level, e.g., within a sub-argument at depth≥2. As deep structures seem to be harder to parse, we would expect longer texts to suffer more.

Next, we look at the models' ability to perform *quasi* argumentative component type (QACT) classification: whether they can correctly predict the role of major claim, AC (non-leaf), AC (leaf) and non-AC, as defined in our auxiliary task described in Section 4.1.1, based on the topology of argumentative structures. This evaluates whether the models place sentences properly in the hierarchical structure. Table 3 shows the result. SBERT-SEQTG [MTL] performed the best, significantly outperforming the second-best SBERT-BIAF (F1-macro=.609 vs .601). We now see the gain of training in the MTL setup as all SEQTG [MTL] models produce better hierarchical arrangements of nodes compared to the STL models; the F1-macro when using BERT encoder is .599 vs .592 (not significant) and SBERT .609 vs .596 (significant).

We notice that BIAF works acceptably well (F1-macro of .601) only when paired with the SBERT encoder. When using the BERT encoder, it has great difficulty in producing *any* non-AC nodes at all (Non-AC F1=.058; F1-macro=.493), despite its good performance on individual links. This result seems to suggest that SBERT is a better encoder than BERT for non-fine-tuning models. This also proves the importance of the evaluation of AM models beyond standard metrics, e.g., in terms of their structural properties as we do here. Prediction performance on individual links does not guarantee the quality of the whole structure. Considering the entire situation, SBERT-BIAF is our preferred model because its performance on standard metrics is substantially better than non-biaffine models. It also performs reasonably well on the hierarchical arrangement of nodes.

We next look at the cross-domain performance of the best sentence linking model, namely SBERT-BIAF. It achieves an accuracy of .459 and an F1-macro of .270 for the prediction of individual links. The F1-macro for QACT classification is .565. These scores are somewhat lower compared to the in-domain performance (significant difference). This means that the modifications of even

dom initialisation in neural networks.

| Model | Major claim | AC (non-leaf) | AC (leaf) | non-AC | F1-macro |
|---|---|---|---|---|---|
| BERT-SEQTG [STL] | .695 | .603 | .584 | .486 | .592 |
| BERT-SEQTG [MTL] | .704 | .594 | .592 | **.507**† | .599 |
| BERT-BIAF | **.730** | .609 | .573 | .058 | .493 |
| SBERT-SEQTG [STL] | .705 | .616 | .590 | .471 | .596 |
| SBERT-SEQTG [MTL] | .725 | .622 | **.611**† | .477 | **.609**† |
| SBERT-BIAF | **.730** | **.639**† | .599 | .437 | .601 |

Table 3: In-domain results of *quasi* argumentative component type classification (node labels identified by topology). We show F1 score per node label and F1-macro. **Bold-face**, †, and underline as above.

25% of essays (in terms of rearrangement) in the out-of-domain data may greatly affect the linking performance, in the cross-domain setting.

### 5.2 Relation Labelling

| | Sup | Det | Att | Res | F1-m |
|---|---|---|---|---|---|
| (B)-FFCON | .698 | .433 | .282 | .594 | .502 |
| (B)-FFLSTM | .695 | .434 | .277 | .600 | .502 |
| (S)-FFCON | .719 | .479 | .372 | .558 | .532 |
| (S)-FFLSTM | .722 | **.481** | .396 | .574 | .543 |
| DISTILBERT | .741 | .426 | .431 | .631 | .557 |
| BERT | **.760**† | .468 | **.478**† | **.673**† | **.595**† |

Table 4: In-domain relation labelling results, showing F1 score per class and F1-macro. "(B)" for BERT and "(S)" for SBERT. **Bold-face**, underline and † as above.

Table 4 shows our experimental results for the in-domain relation labelling task, when gold-standard links are used. BERT model achieves the significantly best performance (F1-macro = .595). Non-fine-tuning models performed better when using SBERT than BERT encoder (F1-macro=.532 vs. .502; .543 vs. .502; both having significant difference). This further confirms the promising potential of SBERT and might suggest that the NLI task is suitable for pre-training a relation labelling model; we plan to investigate this further.

We can see from the results that the ATTACK label is the most difficult one to predict correctly, presumably due to its infrequent occurrence. However, the RESTATEMENT label, which is also infrequent, is relatively well predicted by all models. We think that has to do with all models' ability to recognise semantic similarity. Recall that the RESTATEMENT label is used when a concluding statement rephrases the major claim. SUPPORT and DETAIL are often confused. Note that they are also the most confusing labels between human annotators. Sentence pairs that should be classified as having ATTACK and RESTATEMENT labels are also often classified as SUPPORT.

We also performed our cross-domain experiment for this task. Our best relation labelling model, BERT, achieves the cross-domain F1-macro of .587 (the difference is not significant to in-domain performance). Although not currently shown, the change of performance in other models are also almost negligible (up to 2% in F1-macro).

### 5.3 End-to-end Evaluation

For end-to-end evaluation, we combine in a pipeline system the best models for each task: SBERT-BIAF for sentence linking and fine-tuned BERT for relation labelling.

| | Accuracy | ACI | SL | RL |
|---|---|---|---|---|
| Human-human (IAA) | .474 | .66 | .53 | .61 |
| In-domain | .341 | .42 | .41 | .43 |
| Cross-domain | .321 | .36 | .40 | .39 |

Table 5: End-to-end results. $\kappa$ scores are used for "ACI" (argument component identification), "SL" (sentence linking) and "RL" (relation labelling).

Table 5 shows the evaluation results of the average of 20 runs. Accuracy measures whether the pipeline system predicts all of the following correctly for each source sentence in the text: the correct ACI label (AC vs. non-AC), the correct target distance and the correct relation label. In addition, we also calculated the Cohen's $\kappa$ score between the system's output and the gold annotation for annotation subtasks in our scheme.

The accuracy of the in-domain system is .341, and that of the cross-domain system .321 (significant difference). When compared to human performance on all metrics (in the IAA study), there is still a relatively big performance gap. In an end-to-end setting, the cross-domain system is able to perform at 94% of the in-domain performance. As we feel that this performance drop might well be acceptable in many real-world applications, this signals the potential of training an AM model for

noisy texts using the annotated corpora for well-written texts alongside those more infrequent annotations for noisy text, at least as long as the genre stays the same.

We conducted an error analysis on some random end-to-end outputs. The system tends to fail to identify the correct major claim when it is not placed at the beginning of the essay. For example, the major claim can be pushed into the middle of the essay when an essay contains a lot of background about the discussion topic. Cultural preferences might also play a role. In writings by Asian students, it has been often observed that reasons for a claim are presented before, not after the claim as is more common in anglo-Saxon cultures (Kaplan, 1966; Silva, 1993; Connor, 2002) (as illustrated in Figure 1). The BiLSTM-based models, which are particularly sensitive to order, can be expected to be thrown off by such effects.

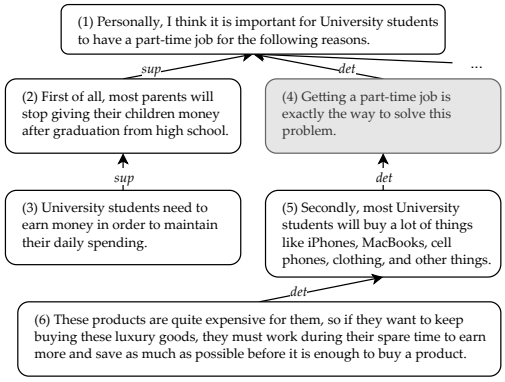



Figure 7: An example snippet of the in-domain system output for the essay code "W_HKG_PTJ0_021_B1_1."

Another source of error concerns placing a sub-argument into the main argument's sibling position instead of that of its child. In general, the systems also have some problems to do with clustering, i.e, splitting a group of sentences that should belong together into separate sub-arguments, or reversely, grouping together sentences that do not belong together. Thus, in order to move forward, the system needs improvement concerning the hierarchical arrangement of sentences in the structure. Figure 7 illustrates this problem. In the gold structure, sentence (4) points at (2), forming a sub-argument (sub-tree) of $\{2, 3, 4\}$. However, the system puts sentence (4) in the inappropriate sub-tree. This kind of cases often happens at group boundaries.

We also found that the system may erroneously use the RESTATEMENT label when connecting claims (at depth $= 1$) and major claims, when the claims include almost all tokens that present in the major claim. We suspect that our model learned to depend on lexical overlaps to recognise RESTATEMENT as this type of relation concerns paraphrasing. However, we cannot perform an error analysis to investigate to what extent this has affected the performance on each of the other relation labels, which concern entailment and logical connections.

## 6 Conclusion

This paper presents a study on parsing argumentative structure in the new domain of EFL essays, which are noisy by nature. We used a pipelined neural approach, consisting of a sentence linking and a relation labelling module. Experimental result shows that the biaffine model combined with the SBERT encoder achieved the best overall performance in the sentence linking task (F1-macro of .323 on individual links). We also investigated MTL, which improved the sequence tagger model in certain aspects. In the sentence linking task, we observed that all models produced more meaningful structures when using SBERT encoder, demonstrating its potential for downstream tasks. In the relation labelling task, non-fine tuning models also performed better when using SBERT encoder. However, the best performance is achieved by a fine-tuned BERT model at F1-macro of .595.

We also evaluated our AM parser on a cross-domain setting, where training is performed on both in-domain (noisy) and out-of-domain (cleaner) data, and evaluation is performed on the in-domain test data. We found that the best cross-domain system achieved 94% (Acc of .321) of the in-domain system (Acc of .341) in terms of end-to-end performance. This signals the potential to use well-written texts, together with noisy texts, to increase the size of AM training data. The main challenge of argument parsing lies in the sentence linking task: the model seems to stumble when confronted with the hierarchical nature of arguments, and we will further tackle this problem in the future.

## Acknowledgements

This work was partially supported by Tokyo Tech World Research Hub Initiative (WRHI), JSPS KAKENHI grant number 20J13239 and Support Centre for Advanced Telecommunication Technology Research. We would like to thank anonymous reviewers for their useful and detailed feedback.

## References


Khalid Al-Khatib, Henning Wachsmuth, Johannes Kiesel, Matthias Hagen, and Benno Stein. 2016. A news editorial corpus for mining argumentation strategies. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3433–3443, Osaka, Japan. The COLING 2016 Organizing Committee.

Kevin D. Ashley. 1990. *Modeling legal argument - reasoning with cases and hypotheticals*. Artificial intelligence and legal reasoning. MIT Press.

Nahla Nola Bacha. 2010. Teaching the academic argument in a university efl environment. *Journal of English for Academic Purposes*, 9(3):229 – 241.

Betty Bamberg. 1983. What makes a text coherent. *College Composition and Communication*, 34(4):417–429.

Elena Cabrio and Serena Villata. 2012. Combining textual entailment and argumentation theory for supporting online debates interactions. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 208–212, Jeju Island, Korea. Association for Computational Linguistics.

Winston Carlile, Nishant Gurrapadi, Zixuan Ke, and Vincent Ng. 2018. Give me more feedback: Annotating argument persuasiveness and related attributes in student essays. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 621–631, Melbourne, Australia. Association for Computational Linguistics.

Y. Chu and T. Liu. 1965. On the shortest arborescence of a directed graph.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark. Association for Computational Linguistics.

Ulla Connor. 2002. New directions in contrastive rhetoric. *TESOL Quarterly*, 36(4):493–510.

Mamoru Deguchi and Kazunori Yamaguchi. 2019. Argument component classification by relation identification by neural network and TextRank. In *Proceedings of the 6th Workshop on Argument Mining*, pages 83–91, Florence, Italy. Association for Computational Linguistics.

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, Minneapolis, Minnesota. Association for Computational Linguistics.

Timothy Dozat and Christopher D. Manning. 2017. Deep biaffine attention for neural dependency parsing. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Jack Edmonds. 1967. Optimum branchings. *Journal of Research of the National Bureau of Standards - B. Mathematics and Mathematical Physics*, 71B:233–240.

Steffen Eger, Johannes Daxenberger, and Iryna Gurevych. 2017. Neural end-to-end learning for computational argumentation mining. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11–22, Vancouver, Canada. Association for Computational Linguistics.

Ronald Aylmer Fisher. 1937. *The design of experiments*. Oliver and Boyd, Edinburgh.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew E. Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. Allennlp: A deep semantic natural language processing platform. *CoRR*, abs/1803.07640.

Debanjan Ghosh, Aquila Khanam, Yubo Han, and Smaranda Muresan. 2016. Coarse-grained argumentation features for scoring persuasive essays. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 549–554, Berlin, Germany. Association for Computational Linguistics.

Barbara J. Grosz and Candace L. Sidner. 1986. Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12(3):175–204.

Ivan Habernal and Iryna Gurevych. 2017. Argumentation mining in user-generated web discourse. *Computational Linguistics*, 43(1):125–179.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Eduard H. Hovy. 1991. *Approaches to the Planning of Coherent Text*, pages 83–102. Springer US, Boston, MA.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF models for sequence tagging. *CoRR*, abs/1508.01991.

Ryu Iida and Takenobu Tokunaga. 2014. Building a corpus of manually revised texts from discourse perspective. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 936–941, Reykjavik, Iceland. European Language Resources Association (ELRA).

Roz Invanic. 2004. Discourses of writing and learning to write. *Language and Education*, 18:220–245.

Shinichiro Ishikawa. 2013. The icnale and sophisticated contrastive interlanguage analysis of asian learners of english. *Learner Corpus Studies in Asia and the World*, 1:91–118.

Shinichiro Ishikawa. 2018. The icnale edited essays: A dataset for analysis of l2 english learner essays based on a new integrative viewpoint. *English Corpus Linguistics*, 25:1–14.

Ann M. Johns. 1986. The esl student and the revision process: Some insights from schema theory. *Journal of Basic Writing*, 5(2):70 – 80.

Robert B. Kaplan. 1966. Cultural thought patterns in inter-cultural education. *Language Learning*, 16(1-2):1–20.

Alex Kendall, Yarin Gal, and Roberto Cipolla. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Diederik P. Kingma and Jimmy Lei Ba. 2015. Adam: a method for stochastic optimization. In *Proceedings of International Conference on Learning Representations (ICLR)*.

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.

Marco Lippi and Paolo Torroni. 2016. Argumentation mining: State of the art and emerging trends. *ACM Trans. Internet Technol.*, 16(2).

Makoto Miwa and Mohit Bansal. 2016. End-to-end relation extraction using LSTMs on sequences and tree structures. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1105–1116, Berlin, Germany. Association for Computational Linguistics.

Gaku Morio and Katsuhide Fujita. 2018. End-to-end argument mining for discussion threads based on parallel constrained pointer architecture. In *Proceedings of the 5th Workshop on Argument Mining*, pages 11–21, Brussels, Belgium. Association for Computational Linguistics.

Gaku Morio, Hiroaki Ozaki, Terufumi Morishita, Yuta Koreeda, and Kohsuke Yanai. 2020. Towards better non-tree argument mining: Proposition-level biaffine parsing with task-specific parameterization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3259–3266, Online. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Andreas Peldszus and Manfred Stede. 2013. From argument diagrams to argumentation mining in texts: A survey. *International Journal of Cognitive Informatics and Natural Intelligence*, 7(1):1–31.

Andreas Peldszus and Manfred Stede. 2016. An annotated corpus of argumentative microtexts. In *Argumentation and Reasoned Action - Proceedings of the 1st European Conference on Argumentation, Lisbon, 2015*.

Isaac Persing, Alan Davis, and Vincent Ng. 2010. Modeling organization in student essays. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 229–239, Cambridge, MA. Association for Computational Linguistics.

Jan Wira Gotama Putra, Simone Teufel, Kana Matsumura, and Takenobu Tokunaga. 2020. TIARA: A tool for annotating discourse relations and sentence reordering. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6912–6920, Marseille, France. European Language Resources Association.

Ella Rabinovich, Sergiu Nisioi, Noam Ordan, and Shuly Wintner. 2016. On the similarities between native, non-native and translated texts. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1881, Berlin, Germany. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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 3982–3992.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. In *Proceedings of 5th Workshop on Energy Efficient Machine Learning and Cognitive Computing, NeurIPS*.

Tony Silva. 1993. Toward an understanding of the distinct nature of l2 writing: The esl research and its implications. *TESOL Quarterly*, 27(4):657–677.

Maria Skeppstedt, Andreas Peldszus, and Manfred Stede. 2018. More or less controlled elicitation of argumentative text: Enlarging a microtext corpus via crowdsourcing. In *Proceedings of the 5th Workshop on Argument Mining*, pages 155–163, Brussels, Belgium. Association for Computational Linguistics.

Yi Song, Michael Heilman, Beata Beigman Klebanov, and Paul Deane. 2014. Applying argumentation schemes for essay scoring. In *Proceedings of the First Workshop on Argumentation Mining*, pages 69–78, Baltimore, Maryland. Association for Computational Linguistics.

Christian Stab and Iryna Gurevych. 2014. Annotating argument components and relations in persuasive essays. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1501–1510, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

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

Masaki Uto, Yikuan Xie, and Maomi Ueno. 2020. Neural automated essay scoring incorporating handcrafted features. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6077–6088, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Henning Wachsmuth, Khalid Al-Khatib, and Benno Stein. 2016. Using argument mining to assess the argumentation quality of essays. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1680–1691, Osaka, Japan. The COLING 2016 Organizing Committee.

Bonnie Webber and Aravind Joshi. 2012. Discourse structure and computation: Past, present and future. In *Proceedings of the ACL-2012 Special Workshop on Rediscovering 50 Years of Discoveries*, pages 42–54, Jeju Island, Korea. Association for Computational Linguistics.

Hiroaki Yamada, Simone Teufel, and Takenobu Tokunaga. 2019. Building a corpus of legal argumentation in japanese judgement documents: towards structure-based summarisation. *Artificial Intelligence and Law*, 27(2):141–170.

Fan Zhang and Diane Litman. 2015. Annotation and classification of argumentative writing revisions. In *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 133–143.

## Appendix A. Implementation Notes

**BERT encoder** We use bert-base-multilingual-cased (https://github.com/google-research/bert#pre-trained-models) and bert-as-a-service (https://github.com/hanxiao/bert-as-service). When using BERT, the sentence embedding is created by averaging subword embeddings composing the sentence in question.

**SBERT encoder** We use SBERT model fine-tuned on the NLI dataset ("bert-base-nli-mean-tokens"), https://github.com/UKPLab/sentence-transformers.

**Sequence Tagger** Dropout is applied between each layer, except between encoder and the dimensionality reduction layer because we do not want to lose any embedding information. We train this model using the cross-entropy loss for each prediction layer. The MTL loss is defined as $L = \sum_t \frac{1}{2\sigma_t^2} L_t + \ln(\sigma_t)$, where the loss $L_t$ of each task $t$ is dynamically weighted, controlled by a learnable parameter $\sigma_t$.

**Biaffine** We apply dropout between all layers, following Dozat and Manning (2017). We use the max-Margin criterion to train the biaffine model.

Principally, we can model the whole AM pipeline using the biaffine model by predicting links and their labels at once (e.g., in Morio et al., 2020). This is achieved by predicting another output graph $X \in \mathbb{R}^{N \times N \times L}$, denoting the probability of each node $x_i$ pointing to $x_j$ on a certain relation label $l_i$. However, we leave this as another MTL experiment for future work.

**Relation Labelling Models** We train the relation labelling models with the cross-entropy loss. Dropout is applied between the final dense layer and the prediction layer.

**Hidden Units and Learning Rates** The number of hidden units and learning rates to train our models are shown in Table 6. All models are trained using Adam optimiser (Kingma and Ba, 2015). Our experiment is implemented in PyTorch (Paszke et al., 2019) and AllenNLP (Gardner et al., 2018).

**Hyperparameter Tuning** Before training our models, we first performed the hyperparameter tuning step. To find the best hyperparameter (e.g., batch size, dropout rate, epochs) of each architecture, in combination with each encoder (BERT/SBERT) and each input type (in- or out-of-domain), we perform 5-fold-cross validation on the training set for 5 times, and select the hyperparameter set that produces the best F1-macro score. During the hyperparameter tuning step, we do not coerce the output to form a tree, i.e., only taking the argmax results.

| | Dense1 | LSTM | Dense2 | LR |
|---|---|---|---|---|
| SEQTG | 512 | 256 | 256 | .001 |
| BIAF | 512 | 256 | 256 | .001 |
| FFLSTM | 256 | 128 | 256 | .001 |
| FFCON | 256 | - | 256 | .001 |
| (DISTIL)BERT | - | - | - | $2e^{-5}$ |

Table 6: The number of hidden units and learning rates (LR) of our models. "Dense1" denotes the dimensionality reduction layer (after encoder). "Dense2" denotes the dense layer after BiLSTM (before prediction).