Argument Pair Extraction with Mutual Guidance and Inter-sentence Relation Graph

Jianzhu Bao[1,2], **Bin Liang**[1,2], **Jingyi Sun**[1,2], **Yice Zhang**[1,2]
Min Yang[3], **Ruifeng Xu**[1,4*]
[1]Harbin Institute of Technology (Shenzhen), China
[2]Joint Lab of China Merchants Securities and HITSZ
[3]Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences
[4]Peng Cheng Laboratory, Shenzhen, China
jianzhubao@gmail.com, bin.liang@stu.hit.edu.cn
sunjingyihit@gmail.com, zhangyc_hit@163.com
min.yang@siat.ac.cn, xuruifeng@hit.edu.cn

Abstract

Argument pair extraction (APE) aims to extract interactive argument pairs from two passages of a discussion. Previous work studied this task in the context of peer review and rebuttal, and decomposed it into a sequence labeling task and a sentence relation classification task. However, despite the promising performance, such an approach obtains the argument pairs implicitly by the two decomposed tasks, lacking explicitly modeling of the argument-level interactions between argument pairs. In this paper, we tackle the APE task by a mutual guidance framework, which could utilize the information of an argument in one passage to guide the identification of arguments that can form pairs with it in another passage. In this manner, two passages can mutually guide each other in the process of APE. Furthermore, we propose an inter-sentence relation graph to effectively model the inter-relations between two sentences and thus facilitates the extraction of argument pairs. Our proposed method can better represent the holistic argument-level semantics and thus explicitly capture the complex correlations between argument pairs. Experimental results show that our approach significantly outperforms the current state-of-the-art model.

1 Introduction

Argumentation mining has received increasing research attention in recent years. Existing studies can be categorized into monological argumentation (Stab and Gurevych, 2014; Eger et al., 2017; Potash et al., 2017; Kuribayashi et al., 2019) and dialogical argumentation (Swanson et al., 2015; Morio and Fujita, 2018; Chakrabarty et al., 2019), with the former identifying the argumentation structure of a single monological document, and the latter focusing on the analysis of argumentation in debates or discussions.

Argument pair extraction (APE) is a new task within the field of dialogical argumentation, aiming at extracting interactive argument pairs from two argumentative passages of a discussion. Cheng et al. (2020) investigated this task in the context of peer review and rebuttal, as they involve rich argumentative and interactive discussions. An example of APE is shown in Figure 1, where a review passage and its corresponding rebuttal passage are segmented into arguments and non-arguments at sentence level. The arguments in review can form argument pairs with the arguments in rebuttal, according to the points they discuss.

APE is a highly challenging task because we need to understand not only the argumentation structure presented by each side of the discussion, but also the interaction of arguments between the participants. The interactions between arguments can be complicated, for example, one argument may be paired with multiple other arguments, forming one-to-many relations. This task is essential for understanding the structure of dialogical argumentation and can also support other related tasks, such as argument generation (Hua et al., 2019a) and debate summarization (Chowanda et al., 2017). Due to the rich interaction of complex arguments, peer review and rebuttal are perfect resources for APE, and have also been exploited in other tasks (Hua et al., 2019b; Fromm et al., 2020).

Cheng et al. (2020) proposed to tackle APE by decomposing it into a sequence labeling task and a sentence relation classification task, with the first subtask extracting the arguments in each review or rebuttal, and the second subtask determining whether two sentences belong to the same pair of arguments. These two subtasks are jointly optimized within a multi-task learning framework, and

*Corresponding Author

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 3923–3934
November 7–11, 2021.

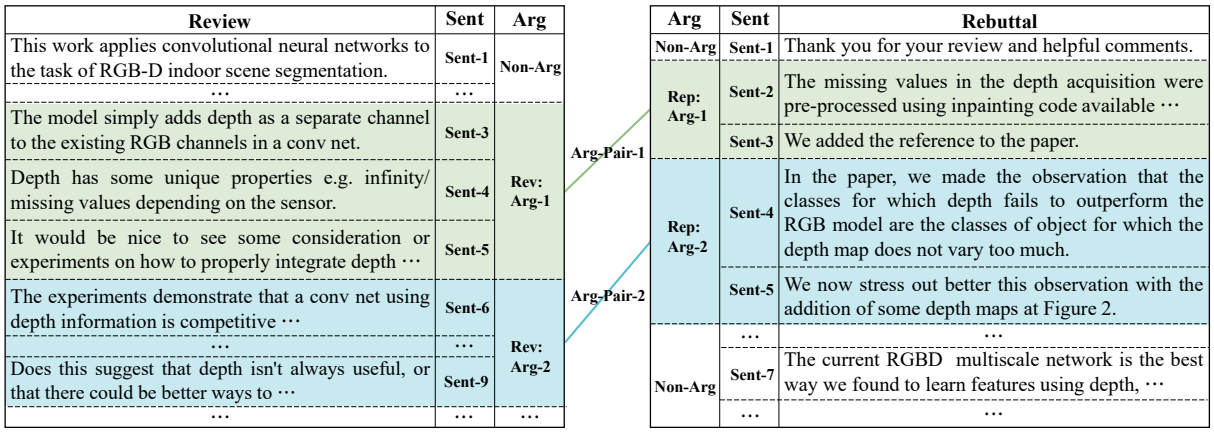

Figure 1: An example of APE. A review passage is shown on the left, and its corresponding rebuttal passage is shown on the right. Sent-i denotes the i-th sentence in the review/rebuttal, and Rev:Arg-i/Rep:Arg-i denotes the i-th argument in the review/rebuttal. Each argument consists of one or more consecutive sentences. Arg-Pair-i denotes the i-th argument pair. In this example, two argument pairs are colored in green and blue respectively.

then the argument pairs are obtained indirectly by combining the results of the two subtasks. However, this method is suboptimal for APE, because it lacks explicitly modeling of the argument-level interactive relations between argument pairs, and the two subtasks might not adapt well to each other.

When humans perform this task, we will first identify an argument from the review passage. Then, keeping this argument in mind, we would further seek out the corresponding arguments from the rebuttal passage to obtain argument pairs. Similarly, this process can be reversed, i.e., we first identify an argument from the rebuttal passage, and then identify the argument in the review passage guided by the identified rebuttal argument. Inspired by this, we design a mutual guidance framework (MGF) to address APE. Our approach first identifies the arguments in the review and rebuttal by a non-guided sequence tagger. Then, incorporating the representations of identified arguments, a review-argument-guided sequence tagger and a rebuttal-argument-guided sequence tagger are utilized to determine argument pairs. Furthermore, we introduce an inter-sentence relation graph (ISRG) to better characterize the complex interactions between review and rebuttal. Unlike the previous method based on two subtasks, our approach can explicitly exploit argument-level semantic information to extract argument pairs more precisely.

Experimental results show that our method significantly outperforms the state-of-the-art methods. Further analysis reveals the effectiveness of mutual guidance and ISRG. Also, our method is more superior when extracting one-to-many pairs.

2 Task Definition

Following the work of Cheng et al. (2020), we aim to automatically extract interactive argument pairs from peer review and rebuttal. Formally, given a review passage $\mathcal{V} = (s_1^v, s_2^v, \ldots, s_m^v)$ consisting of m sentences and a rebuttal passage $\mathcal{B} = (s_1^b, s_2^b, \ldots, s_n^b)$ consisting of n sentences, we first need to identify each argument in review and rebuttal, and obtain a review argument spans set $\hat{\mathrm{X}}^v = \{\hat{\alpha}_1^v, \hat{\alpha}_2^v, \ldots\}$ and a rebuttal argument spans set $\hat{\mathrm{X}}^b = \{\hat{\alpha}_1^b, \hat{\alpha}_2^b, \ldots\}$, where $\hat{\alpha}_i^v$ and $\hat{\alpha}_i^b$ are sentence-level spans in review and rebuttal, respectively. Then, a set of interactive argument pairs $\hat{\mathrm{P}} = \{\hat{p}_1, \hat{p}_2, \ldots\}$ should be extracted, where $\hat{p}_i \in \hat{\mathrm{X}}^v \times \hat{\mathrm{X}}^b$ is an interactive argument pair. For example, in Figure 1, the review argument spans set $\hat{\mathrm{X}}$ is $\{\hat{a}_1^v, \hat{a}_2^v\} = \{(3,5),(6,9)\}$ and the rebuttal argument spans set $\hat{\mathrm{Y}}$ is $\{\hat{a}_1^b, \hat{a}_2^b\} = \{(2,3),(4,5)\}$. The argument pairs set $\hat{\mathrm{P}}$ is $\{(\hat{a}_1^v, \hat{a}_1^b),(\hat{a}_2^v, \hat{a}_2^b)\}$.

3 Proposed Approach

We present a mutual guidance framework with an inter-sentence relation graph for APE, named MGF. Our approach can better utilize the holistic argument-level semantics and thus explicitly capture the complex correlations between argument pairs. The overall architecture is shown in Figure 2. In the following, we first introduce the inter-sentence relation graph, then describe the mutual guidance framework.

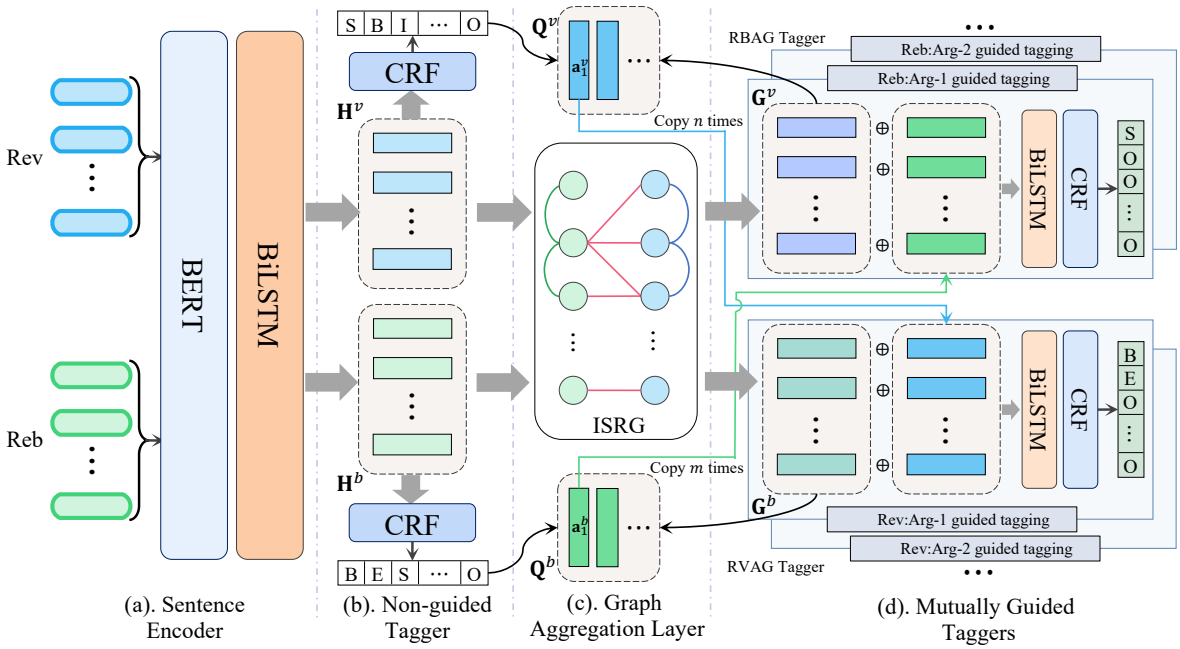

Figure 2: The architecture of MGF.

3.1 Inter-sentence Relation Graph

In order to facilitate argument pair extraction, we capture the latent sentence relations between review and rebuttal by an inter-sentence relation graph. This graph regards every sentence in review and rebuttal as nodes, and is constructed from two perspectives: 1) From the in-passage perspective, we build edges among the sentences of individual review/rebuttal passage (in-passage edges) based on the relative positions between them. This kind of edge can emphasize the correlation between two sentences with close distance, as they may be in the same argument. 2) From the cross-passage perspective, we build edges between review sentences and rebuttal sentences (cross-passage edges) based on the co-occurring words between two sentences. Intuitively, two arguments in an argument pair are likely to share certain words since they are discussing the same point. Also, we find that there are co-occurring words in more than 80% of the argument pairs of the Review-Rebuttal dataset (Cheng et al., 2020) (ignoring the stop words). Thus, this kind of edge could help capture the interactions between argument pairs by modeling the cross-passage sentence relations.

In-passage Edge. Based on the relative positions between two sentences, the weights of the edge between every two in-passage sentences $\omega^I(s_i, s_j)$ can be computed as:

$$\omega^I(s_i, s_j) = \begin{cases} 1 + (1 - \frac{\mathcal{D}(s_i,s_j)}{\rho}) & \mathcal{D}(s_i, s_j) \leq \rho \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where s_i and s_j are two sentences within an individual review/rebuttal passage, and $\mathcal{D}(s_i, s_j)$ denotes the relative distance between them. ρ is the in-passage sentence distance threshold, and two sentences are connected only if their relative distance is not greater than ρ. Since most passages are very long, this threshold ρ can control the farthest retention distance, so as to reduce noise.

Cross-passage Edge. Based on the co-occurring words between two sentences, the weights of the edge between every two cross-passage sentences $\omega^C(s_i, s_j)$ can be computed as:

$$\omega^C(s_i, s_j) = \begin{cases} 1 + \frac{\mathcal{C}(s_i,s_j)}{\mathcal{C}_{max}} & \mathcal{C}(s_i, s_j) > \varphi \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

where s_i and s_j are two sentences from two different passages, and $\mathcal{C}(s_i, s_j)$ denotes the number of co-occurring words of them. $\mathcal{C}_{max}$ is the maximum co-occurring words number of the corpus. φ indicates the co-occurring words number threshold, and two passage sentences are connected only when the number of their co-occurring words is greater than φ. Note that when calculating $\mathcal{C}(s_i, s_j)$, we ignore the stop words.

With the in-passage edges and the cross-passage edges defined above, the inter-sentence relation graph (ISRG) of review $\mathcal{V}$ and rebuttal $\mathcal{B}$ could be constructed, where the nodes are all sentences of review and rebuttal. Here, the adjacency matrix $\mathbf{A} \in \mathbb{R}^{(m+n)\times(m+n)}$ of ISRG can be derived as:

$$\mathrm{A}_{ij} = \begin{cases} \omega^I(s_i, s_j) & s_i, s_j \in \mathcal{V} \\ \omega^I(s_i, s_j) & s_i, s_j \in \mathcal{B} \\ \omega^C(s_i, s_j) & s_i \in \mathcal{V}, s_j \in \mathcal{B} \\ \omega^C(s_i, s_j) & s_i \in \mathcal{B}, s_j \in \mathcal{V} \end{cases} \quad (3)$$

3.2 Mutual Guidance Framework

Our proposed Mutually Guided Framework (MGF) first encodes the sentences and employs a non-guided sequence tagger to identify the arguments in the review and rebuttal. Then, after obtaining a relation-oriented sentence representation by graph convolution, two mutually guided taggers are used to extract argument pairs.

Sentence Encoder. We apply BERT (Devlin et al., 2019) to obtain the representation of each sentence and use LSTM (Hochreiter and Schmidhuber, 1997) to encode the contextual long-term dependencies of sentences. Specifically, for each sentence s_i from $\mathcal{V}$ or $\mathcal{B}$, we feed it into BERT and get the sentence embedding $\mathbf{e}_i \in \mathbb{R}^{d_b}$ by mean pooling over all token representations, where d_b is the vector dimension of the last layer of BERT. Hence, the sentences in $\mathcal{V}$ and $\mathcal{B}$ can be represented as $\mathbf{V} = (\mathbf{e}_1^v, \mathbf{e}_2^v, \ldots, \mathbf{e}_m^v)$ and $\mathbf{B} = (\mathbf{e}_1^b, \mathbf{e}_2^b, \ldots, \mathbf{e}_n^b)$. Subsequently, $\mathbf{V}$ and $\mathbf{B}$ are separately fed into a bidirectional LSTM (BiLSTM), and the hidden states from both directions of each sentence are concatenated as the contextual sentence representation. In this way, the contextual sentence representation matrix of $\mathcal{V}$ and $\mathcal{B}$ can be derived:

$$\mathbf{H}^v = (\mathbf{h}_1^v, \mathbf{h}_2^v, \ldots, \mathbf{h}_m^v) \quad (4)$$

$$\mathbf{H}^b = (\mathbf{h}_1^b, \mathbf{h}_2^b, \ldots, \mathbf{h}_n^b) \quad (5)$$

where $\mathbf{h}_i^v/\mathbf{h}_i^v \in \mathbb{R}^{2d_l}$ is the contextual sentence representation of the i-th sentence in review/rebuttal, d_l is the hidden size of LSTM.

Non-guided Tagger. We use a CRF sequence tagger to identify all potential arguments, named non-guided tagger, which could provide explicit argument span information for the subsequent argument pairs extraction. Concretely, we feed the contextual sentence representations $\mathbf{H}^v$ and $\mathbf{H}^b$ into this CRF tagger, and the predicted label sequences for review and rebuttal could be obtained:

$$\mathrm{Y}^v = (y_1^v, y_2^v, \ldots, y_m^v) \quad (6)$$

$$\mathrm{Y}^b = (y_1^b, y_2^b, \ldots, y_n^b) \quad (7)$$

where y_i^v/y_i^b is the IOBES label for the i-th sentence of review/rebuttal.

According to these two label sequences, we could obtain the potential argument spans for review and rebuttal, i.e. $\mathrm{X}^v = \{\alpha_1^v, \alpha_2^v, \ldots\}$ and $\mathrm{X}^b = \{\alpha_1^b, \alpha_2^b, \ldots\}$, where α_i^v/α_i^b is the i-th predicted argument span of review/rebuttal.

Graph Aggregation Layer. Base on the inter-sentence relation graph constructed in Section 3.1, we use the contextual sentence representations $\mathbf{H}^v \in \mathbb{R}^{m\times 2d_l}$ and $\mathbf{H}^b \in \mathbb{R}^{n\times 2d_l}$ as the feature vectors of $(m+n)$ nodes in this graph. Then, we employ a graph convolutional network (GCN) (Kipf and Welling, 2017) to conduct information exchange between nodes:

$$\mathbf{G}^{(0)} = \left[\mathbf{H}^v; \mathbf{H}^b\right] \quad (8)$$

$$\mathbf{G}^{(l+1)} = \sigma(\widetilde{\mathbf{A}}\mathbf{G}^{(l)}\mathbf{W}^{(l)} + \mathbf{b}^{(l)}) \quad (9)$$

where $\mathbf{G}^l \in \mathbb{R}^{(m+n)\times 2d_l}$ contains all node vectors in the l-th layer of GCN and $\widetilde{\mathbf{A}}$ is the normalized adjacency matrix. $\mathbf{W}^{(l)}$ and $\mathbf{b}^{(l)}$ are learnable parameter matrix and bias term. $\sigma(\cdot)$ is the ReLU activation function commonly used in GCN.

We keep the node vectors of the last layer of the GCN as the relation-oriented sentence representations of sentences for review ($\mathbf{G}^v$) and rebuttal ($\mathbf{G}^b$):

$$\mathbf{G}^{(L)} = \left[\mathbf{G}^v; \mathbf{G}^b\right] \quad (10)$$

$$\mathbf{G}^v = (\mathbf{g}_1^v, \mathbf{g}_2^v, \ldots, \mathbf{g}_m^v) \quad (11)$$

$$\mathbf{G}^b = (\mathbf{g}_1^b, \mathbf{g}_2^b, \ldots, \mathbf{g}_n^b) \quad (12)$$

where $\mathbf{g}_i^v/\mathbf{g}_i^b \in \mathbb{R}^{d_g}$ is the relation-oriented representation for the i-th sentence in review/rebuttal, and d_g is the output feature dimension of GCN.

Mutually Guided Taggers. With the argument spans sets (X^v and X^b) produced by the non-guided tagger and the relation-oriented sentence representations ($\mathbf{G}^v$ and $\mathbf{G}^b$) produced by GCN, we could extract argument pairs with two mutually guided taggers, i.e. review-argument-guided (RVAG) tagger and rebuttal-argument-guided (RBAG) tagger. These two taggers could guide each other and cooperate to extract argument pairs.

For the RVAG tagger, we first use review argument spans set X^v to produce a representation of each potential review argument from $\mathbf{G}^v$ by mean pooling over the sentence representations in each argument span. Specifically, for the k-th argument span $\alpha_k^v = (b_k, e_k)$ in X^v, the contextual representation of this argument $\mathbf{a}_k^v \in \mathbb{R}^{d_g}$ could be obtained by:

$$\mathbf{a}_k^v = \frac{1}{e_k - b_k + 1} \sum_{i=b_k}^{e_k} \mathbf{g}_i^v \tag{13}$$

In this way, the representations of review arguments can be represented as $\mathbf{Q}^v = (\mathbf{a}_1^v, \mathbf{a}_2^v, \dots)$. Subsequently, to enable this k-th review argument to guide the identification of its paired rebuttal arguments, we concatenate $\mathbf{a}_k^v$ to each rebuttal sentence representation $\mathbf{g}_i^b$ and then apply another BiLSTM to obtain the RVAG rebuttal sentence representations:

$$\overrightarrow{\mathbf{h}}_i^{b,g} = \overrightarrow{\mathrm{LSTM}}(\mathbf{g}_i^b \oplus \mathbf{a}_k^v, \overrightarrow{\mathbf{h}}_{i-1}^{b,g}) \tag{14}$$

$$\overleftarrow{\mathbf{h}}_i^{b,g} = \overleftarrow{\mathrm{LSTM}}(\mathbf{g}_i^b \oplus \mathbf{a}_k^v, \overleftarrow{\mathbf{h}}_{i-1}^{b,g}) \tag{15}$$

$$\mathbf{h}_i^{b,g} = \overrightarrow{\mathbf{h}}_i^{b,g} \oplus \overleftarrow{\mathbf{h}}_i^{b,g} \tag{16}$$

where $\mathbf{h}_i^{b,g} \in \mathbb{R}^{d_l}$ is the RVAG representations for the i-th sentence in rebuttal. In this way, the RVAG rebuttal sentence representation matrix $\mathbf{H}^{v,g} = (\mathbf{h}_1^{b,g}, \mathbf{h}_2^{b,g}, \dots, \mathbf{h}_n^{b,g})$ could be obtained. Then, we input $\mathbf{H}^{v,g}$ into a CRF layer to identify the arguments that could form pairs with the k-th review argument α_k^v.

Similarly, the RBAG tagger can be conducted in the same manner, except that each identified rebuttal argument is used to guide the identification of its paired review arguments.

3.3 Training

The loss function of MGF consists of two parts, one for AM and the other for APE.

For AM, we maximize the log-likelihood of the non-guided tagger:

$$\mathcal{L}_{am} = \log p(\hat{\mathrm{Y}}^v|\mathcal{V}) + \log p(\hat{\mathrm{Y}}^b|\mathcal{B}) \tag{17}$$

where $\hat{\mathrm{Y}}^v$ and $\hat{\mathrm{Y}}^b$ are the ground-truth IOBES label sequences of the review and rebuttal.

For APE, the log-likelihood of the RVAG tagger and the RBAG tagger are as follows:

$$\begin{aligned}\mathcal{L}_{ape} = &\sum_i \log p(\hat{\mathrm{Y}}_i^{b,r}|\mathcal{B}, \mathrm{X}^v) \\ &+ \sum_i \log p(\hat{\mathrm{Y}}_i^{v,r}|\mathcal{V}, \mathrm{X}^b)\end{aligned} \tag{18}$$

where $\hat{\mathrm{Y}}_i^{v,r}$ and $\hat{\mathrm{Y}}_i^{b,r}$ are the i-th relation-oriented ground-truth IOBES label sequences of review and rebuttal. Concretely, all review arguments derived by the label sequence $\hat{\mathrm{Y}}_i^{v,r}$ are paired with the i-th argument of the rebuttal.

We sum the loss function of the above two parts to obtain the final training objective of MGF[1]:

$$\mathcal{L} = \mathcal{L}_{am} + \mathcal{L}_{ape} \tag{19}$$

3.4 Inference

During inference, we fuse the prediction of both RVAG tagger and RBAG tagger to obtain argument pairs. Specifically, let $\mathrm{Y}_k^{v,r}$ denote the relation-oriented label sequences predicted by the RBAG tagger, from which all review argument spans paired with the k-th rebuttal argument can be obtained. We notate these review argument spans as $\mathrm{X}_k^{v,r} = (\alpha_{k,1}^v, \alpha_{k,2}^v, \dots)$ and the k-th rebuttal argument span as α_k^b. Accordingly, the argument pairs derived from $\mathrm{Y}_k^{v,r}$ can be denoted as $\mathrm{P}_k^{v,r} = ((\alpha_{k,1}^v, \alpha_k^b), (\alpha_{k,2}^v, \alpha_k^b), \dots)$. Further, we can obtain all argument pairs predicted by RBAG tagger P^{rbag} by:

$$\mathrm{P}^{rbag} = \bigcup_k \mathrm{P}_k^{v,r} \tag{20}$$

Similarly, all argument pairs predicted by RVAG tagger P^{rvag} can be obtained in the same manner.

Then, we consider the union set of P^{rvag} and P^{rbag} as the prediction result of argument pairs, i.e. $\mathrm{P} = \mathrm{P}^{rvag} \cup \mathrm{P}^{rbag}$. Our preliminary experimental results show that this approach can efficiently fuse the prediction results of RVAG tagger and RBAG tagger.

4 Experimental Setup

4.1 Dataset

We conduct experiments on the Review-Rebuttal (RR) dataset proposed by Cheng et al. (2020). This dataset contains 4,764 review-rebuttal passage pairs of ICLR collected from openreview.net. Cheng et al. (2020) provided two versions of dividing RR dataset, namely RR-submission and RR-passage. In both versions, RR dataset is split by the ratio of 8:1:1 for training, development, and testing. In RR-submission, multiple review-rebuttal passage pairs of the same paper submission are in

[1]We considered putting different weights for these two parts, but the impact is minimal. Detailed experimental results can be found in Appendix A.

RR	# Review-rebuttal pairs	4,764
	# Argument pairs	18.6K
	# One-to-one argument pairs	13.0K
	# One-to-many argument pairs	5.6K
Rev	# Sentences	99.8K
	# Arguments	23.2K
	Avg. # sentences per passage	21.0
	Avg. # sentences per argument	2.5
Reb	# Sentences	94.9K
	# Arguments	17.7K
	Avg. # sentences per passage	19.9
	Avg. # sentences per argument	3.8

Table 1: Statistics of RR dataset.

the same train/development/test set, whereas RR-passage does not guarantee this. This distinction makes RR-submission more challenging, so our further experiments are conducted on RR-submission. The detailed statistics about RR dataset are summarized in Table 1.

4.2 Implementation Details

We evaluate our experiments by two metrics, namely argument mining (AM) and argument pair extraction (APE). Unlike Cheng et al. (2020), we do not use sentence pairing as an evaluation metric since we extract argument pairs directly instead of using sentence pairing as a subtask. We employ the precision (Pre.), recall (Rec.), and F_1 scores to measure the performance on AM and APE. All experiments are performed 5 times with different random seeds, and the scores are averaged.

Regarding the implementation of our model[2], we adopt the uncased BERT_{Base} [3] as our base encoder, which is fine-tuned during training. All LSTMs used in our model are 1 layer with the hidden size of 512. Note that, the parameters of LSTMs and CRFs used in the three taggers are not shared. The AdamW optimizer (Kingma and Ba, 2015) is employed for parameter optimization, and the initial learning rates for BERT layer and other layers are set to 1e-5 and 1e-3, respectively. The dropout rate (Srivastava et al., 2014) is set to 0.5 and the batch size is 2. Our model is implemented in PyTorch (Paszke et al., 2019) on a NVIDIA Tesla V100 GPU. We train our model 10 epochs with early stopping strategy, and choose the best model parameters based on the best performance on the development set (average of F_1 score of AM and APE).

[2]Our source code is available at https://github.com/HLT-HITSZ/MGF.

[3]https://github.com/huggingface/transformers

4.3 Baselines

To evaluate our mutual guidance framework (MGF), we compare it with several baselines:

PL-H-LSTM-CRF (Cheng et al., 2020) independently trains a sequence labeling model and a sentence relation classification model, and then pipes the result together to obtain argument pairs.

MT-H-LSTM-CRF (Cheng et al., 2020) is similar to PL-H-LSTM-CRF, except that it trains two subtasks in a multi-task framework. This is the current state-of-the-art method on RR dataset. Note that the BERT encoder used in this model is not fine-tuned during training.

Besides, we implemented two additional baselines for further comparisons:

Two-Step is another pipeline model. Unlike PL-H-LSTM-CRF, this model first identifies all potential arguments by sequence labeling, then matches review arguments and rebuttal arguments by Cartesian products to determine argument pairs. Both steps are based on BERT.

Non-FT-MGF is the implementation of our framework based on the sentence encoding method of MT-H-LSTM-CRF. It does not fine-tune BERT for a fair comparison with MT-H-LSTM-CRF.

5 Results and Analysis

5.1 Main Results

The overall performance of our proposed framework and the baselines are shown in Table 2. Our model achieves the best performance on both RR-submission and RR-passage. On RR-submission, our model outperforms the current state-of-the-art model MT-H-LSTM-CRF by at least 1.01% and 7.94% in F_1 score over AM and APE. On RR-passage, our model also outperforms MT-H-LSTM-CRF and obtains at least 0.79% and 7.01% higher F_1 scores over AM and APE.

We also show the results where the sentence encoder of MGF is replaced by that of MT-H-LSTM-CRF, namely Non-FT-MGF. Without employing BERT fine-tuning, Non-FT-MGF still outperforms MT-H-LSTM-CRF, which demonstrates that the performance gains we achieve are not solely due to BERT fine-tuning. It can also be observed that our model results can be further improved with BERT fine-tuning by comparing MGF with Non-FT-MGF.

Data	Method	Argument Mining			Argument Pair Extraction		
		Pre.	Rec.	F_1	Pre.	Rec.	F_1
RR-submission	PL-H-LSTM-CRF	67.63	68.51	68.06	19.86	19.94	19.90
	MT-H-LSTM-CRF	70.09	70.14	70.12	26.69	26.24	26.46
	Two-Step	**70.94**	70.77	70.86	33.11	24.67	28.27
	Non-FT-MGF	69.18	69.94	69.55	33.12	33.69	33.40
	MGF (Ours)	70.40	**71.87**	**71.13**	**34.23**	**34.57**	**34.40**
RR-passage	PL-H-LSTM-CRF	73.10	67.65	70.27	21.24	19.30	20.23
	MT-H-LSTM-CRF	71.85	71.01	71.43	30.08	29.55	29.81
	Two-Step	71.94	**71.51**	71.72	34.31	26.87	30.14
	Non-FT-MGF	71.22	70.49	70.85	35.20	34.11	34.65
	MGF (Ours)	**73.62**	70.88	**72.22**	**38.03**	**35.68**	**36.82**

Table 2: Comparison results with baselines on RR-submission and RR-passage (%). The best scores are in bold.

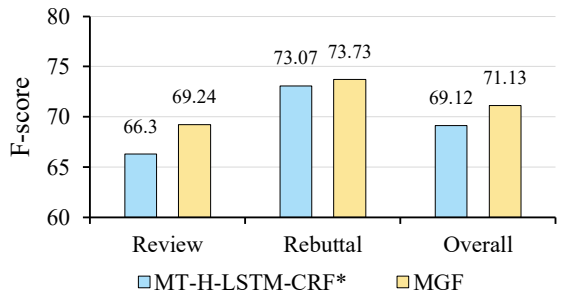

Figure 3: Detailed results of AM (%). * indicates the results we replicated, as the authors of MT-H-LSTM-CRF did not provide these results.

Method	APE F_1	∇
MGF (Ours)	**34.40**	-
w/o RVAG Tagger	33.11	-1.29
w/o RBAG Tagger	31.94	-2.46
w/o ISRG	30.65	-3.75
w/o IPE	33.12	-1.28
w/o CPE	32.33	-2.07

Table 3: The results of ablation experiments on RR-submission (%). The best scores are in bold.

5.2 Detailed Results of Argument Mining

Figure 3 shows the detailed results of AM on RR-submission. Here, we compare the performances of MGF and MT-H-LSTM-CRF on review passages and rebuttal passages, respectively. Since rebuttal passages are more clearly arranged and structured than review passages (Cheng et al., 2020), both models perform better on the former. Although our MGF yielded similar AM results to MT-H-LSTM-CRF on rebuttal passages, it shows significant improvement on more complex review passages.

5.3 Ablation Study

As shown in Table 3, we conduct ablation experiments to further evaluate the contribution of each component in our proposed MGF. The F_1 score decreases heavily without mutual guidance. Specifically, the F_1 score of APE decreases by 2.46% if only RVAG tagger is used (*w/o* RBAG Tagger). Similarly, using only the RBAG tagger (*w/o* RVAG Tagger) decreases the F_1 score by 1.29%. Such results validate the effectiveness of our proposed mutual guidance framework. Furthermore, we can observe that the performance of using only RBAG tagger is better than that of using only RVAG tagger. This is possibly due to the fact that, on the AM task, the identification of the rebuttal arguments is more accurate than the review arguments (Figure 3), leading to better results when using identified rebuttal arguments to guide argument pair extraction.

It can be observed that without our proposed inter-sentence relation graph (*w/o* ISRG), the F_1 score drops heavily (-3.75%). Going one step further, if we exclude the in-passage edges (*w/o* IPE), the F_1 score will decrease by 1.28%, indicating the necessity of capturing interactions between two sentences with close distance. Also, incorporating cross-passage edges into MGF (*w/o* CPE) can bring more significant F_1 score improvement (2.07%), because cross-passage edges can model the sentence relations cross two passages and thus facilitate the identification of interactive argument pairs.

Type of pairs	Method	APE Rec.
All	MT-H-LSTM-CRF*	26.05
	MGF (Ours)	34.57
One-to-one	MT-H-LSTM-CRF*	35.86
	MGF (Ours)	41.37
One-to-many	MT-H-LSTM-CRF*	11.09
	MGF (Ours)	17.71

Table 4: Results of extracting one-to-many pairs on RR-submission (%). Similar to Figure 3, * denotes the results that we replicated.

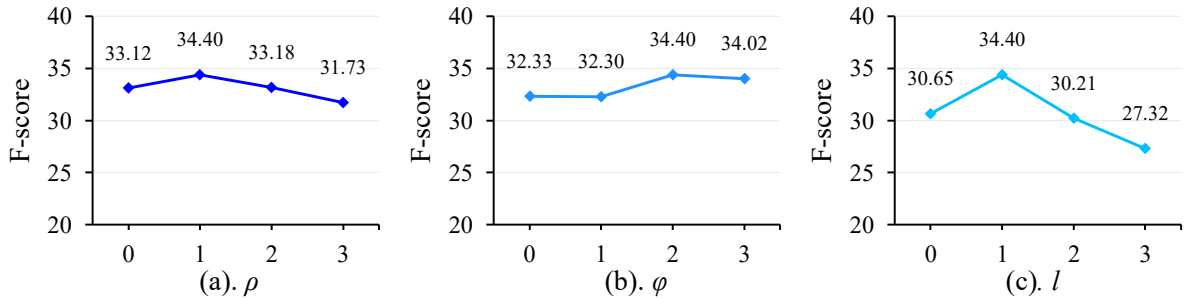

Figure 4: Impacts of graph parameters.

5.4 Results of Extracting One-to-many Pairs

We further compare the results on extracting one-to-many pairs on RR-submission in Table 4. We divide argument pairs of the test set into two subsets: one subset contains only one-to-one argument pairs, and the other subset contains only one-to-many argument pairs. Then, we compare the recall of MT-H-LSTM-CRF and MGF on the two subsets.

It can be seen that our MGF model consistently outperforms MT-H-LSTM-CRF on both subsets. Furthermore, MGF is relatively more effective for one-to-many argument pairs, with a recall improvement of 6.62%. This improvement comes from the ability of our model to take into account the entire review/rebuttal sequence when extracting argument pairs, so that multiple arguments that form pairs with the guiding argument could be extracted simultaneously through sequence tagging.

5.5 Impacts of Graph Parameters

The inter-sentence relation graph for modeling inter-sentence latent relations is a critical part of our model. Therefore, we further investigate the impacts of the graph parameters on the performance of MGF, including the threshold of in-passage sentence distance ρ, the threshold of co-occurring words number φ, and the number of GCN layers l. The detailed results are shown in Figure 4.

From Figure 4(a), our approach achieves the best performance with ρ set to 1. With this setting, each sentence node in the graph is directly connected to the two sentence nodes that are adjacent to it in the passage. Such a phenomenon is consistent with our observation in Table 1 that the average number of sentences contained in each argument is 3.1. Since the majority of arguments contain a small number of sentences, we should not connect two sentences that have a long distance. Otherwise, the semantic representation of arguments will be distorted.

According to Figure 4(b), we find that it is most appropriate to set φ to 2. This suggests that two sentences with more than 2 co-occurring words are more likely to be from two inter-related arguments. If we set φ too small, then too much noise will be introduced. Conversely, if we set φ too large, then many sentence pairs from two inter-related arguments will be ignored by the graph.

For the number of GCN layers l, our approach performs best with 1 layer GCN, indicating that the inter-sentence relations can be modeled sufficiently without stacking many layers of GCN.

5.6 Error Analysis

To gain a deeper insight into our method, we analyze the prediction of our model. To be specific, we randomly sampled 100 samples from the test set of RR-submission, and then manually inspect the prediction results. Here are two major causes of errors.

- It is difficult to extract argument pairs if there are no co-occurring or semantically similar words in two arguments. In this scenario, our proposed ISRG based on co-occurring words cannot provide valid information. Also, it is hard for the pre-trained model to capture the association between such argument pairs.
- In some cases, our model identifies only a few important sentences instead of a complete argument. However, in some other cases, multiple consecutive arguments are identified as one argument. The reason is that we frame both AM and APE as sentence-level sequence tagging tasks. For such a task, the boundaries of arguments are often diverse and difficult to determine, so the model often misidentifies them.

6 Related Work

Most existing studies in the field of argumentaion mining focus on monological argumentation, such as argumentation structure parsing(Stab and

Gurevych, 2017; Afantenos et al., 2018; Kuribayashi et al., 2019; Hua et al., 2019b; Morio et al., 2020), automated essay scoring(Wachsmuth et al., 2016; Ke et al., 2018; Song et al., 2020), argument quality assessment(Wachsmuth et al., 2017; Gretz et al., 2020; Lauscher et al., 2020), argumentation strategies modeling(Khatib et al., 2016, 2017), etc.

Since real-life argumentation is usually in the form of dialogue, some prior work focuses on dialogical argumentation. Morio and Fujita (2018) employed a pointer network to predict argumentation structures in discussion threads. Chakrabarty et al. (2019) studied the relations between argument components in online discussion forums with pre-trained models and discourse relations. Ji et al. (2019) proposed a discrete argument representation learning method to extract argument pairs. However, these studies above assumed that the boundaries of arguments have been given. Recently, Cheng et al. (2020) present a new task named argument pair extraction, which is more challenging as it requires both identifying arguments from plain text and extracting the interactive argument pairs.

Our work is closely related to the argument relation prediction task. Many studies of argumentation structure parsing include argumentative relation prediction as a subtask(Kuribayashi et al., 2019; Morio et al., 2020; Bao et al., 2021). Since argument relation prediction is highly challenging, recently, more and more researchers study it as an independent task(Chen et al., 2018; Opitz and Frank, 2019; Cocarascu et al., 2020; Jo et al., 2021). Despite the strong connection, APE task is more challenging than argument relation prediction. Specifically, in argument relation prediction, arguments are given. But for APE, only two plain documents without any pre-labeled information are given, and we need to identify arguments in two documents and determine argument relations simultaneously.

Graph neural networks (GNN) have shown promising performance in many NLP tasks, such as text classification(Yao et al., 2019; Ragesh et al., 2021), question answering(Tu et al., 2019; Qiu et al., 2019), sentiment analysis(Liang et al., 2021, 2020), text summarization(Xu et al., 2020; Yasunaga et al., 2017), etc. Recently, some works have attempted to introduce GNN into argumentation mining. Morio and Fujita (2019) performed argument component identification and classification by syntactic graph convolutional networks. Huang et al. (2021) proposed a heterogeneous argument attention network for argumentation persuasiveness prediction. In this paper, our proposed inter-sentence relation graph can effectively model the inter-relations between two sentences, thus facilitating APE.

7 Conclusion

In this paper, we propose an effective mutual guidance framework for argument pair extraction, named MGF, which enables arguments of two passages to mutually guide each other for extracting interactive argument pairs. In addition, we introduce an inter-sentence relation graph into our proposed MGF, which could effectively model the inter-relations between two sentences and thus improving the extraction of argument pairs. The experimental results demonstrate the effectiveness of our method. In the future, we plan to apply our method to datasets from more diverse domains beyond the peer review and rebuttal, such as social networks, debate competitions, etc.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China (61632011, 61876053, 62006062, 62176076), the Guangdong Province Covid-19 Pandemic Control Research Funding (2020KZDZX1224), the Shenzhen Foundational Research Funding (JCYJ20180507183527919 and JCYJ20200109113441941), China Postdoctoral Science Foundation (2020M670912), Joint Lab of HITSZ and China Merchants Securities, Youth Innovation Promotion Association of CAS China (No. 2020357), and Shenzhen Science and Technology Innovation Program (Grant No. KQTD20190929172835662).

References

Stergos D. Afantenos, Andreas Peldszus, and Manfred Stede. 2018. Comparing decoding mechanisms for parsing argumentative structures. *Argument Comput.*, 9(3):177–192.

Jianzhu Bao, Chuang Fan, Jipeng Wu, Yixue Dang, Jiachen Du, and Ruifeng Xu. 2021. A neural transition-based model for argumentation mining. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1:*

Long Papers), Virtual Event, August 1-6, 2021, pages 6354–6364. Association for Computational Linguistics.

Tuhin Chakrabarty, Christopher Hidey, Smaranda Muresan, Kathy 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 2019, Hong Kong, China, November 3-7, 2019*, pages 2933–2943. Association for Computational Linguistics.

Di Chen, Jiachen Du, Lidong Bing, and Ruifeng Xu. 2018. Hybrid neural attention for agreement/disagreement inference in online debates. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 665–670. Association for Computational Linguistics.

Liying Cheng, Lidong Bing, Qian Yu, Wei Lu, and Luo Si. 2020. APE: argument pair extraction from peer review and rebuttal via multi-task learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 7000–7011. Association for Computational Linguistics.

Alan Darmasaputra Chowanda, Albert Richard Sanyoto, Derwin Suhartono, and Criscentia Jessica Setiadi. 2017. Automatic debate text summarization in online debate forum. In *ICCSCI*, pages 11–19.

Oana Cocarascu, Elena Cabrio, Serena Villata, and Francesca Toni. 2020. Dataset independent baselines for relation prediction in argument mining. In *Computational Models of Argument - Proceedings of COMMA 2020, Perugia, Italy, September 4-11, 2020*, volume 326 of *Frontiers in Artificial Intelligence and Applications*, pages 45–52. IOS Press.

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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

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, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 11–22. Association for Computational Linguistics.

Michael Fromm, Evgeniy Faerman, Max Berrendorf, Siddharth Bhargava, Ruoxia Qi, Yao Zhang, Lukas Dennert, Sophia Selle, Yang Mao, and Thomas Seidl. 2020. Argument mining driven analysis of peer-reviews. *CoRR*, abs/2012.07743.

Shai Gretz, Roni Friedman, Edo Cohen-Karlik, Assaf Toledo, Dan Lahav, Ranit Aharonov, and Noam Slonim. 2020. A large-scale dataset for argument quality ranking: Construction and analysis. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7805–7813. AAAI Press.

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

Xinyu Hua, Zhe Hu, and Lu Wang. 2019a. Argument generation with retrieval, planning, and realization. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 2661–2672. Association for Computational Linguistics.

Xinyu Hua, Mitko Nikolov, Nikhil Badugu, and Lu Wang. 2019b. Argument mining for understanding peer reviews. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 2131–2137. Association for Computational Linguistics.

Kuo Yu Huang, Hen-Hsen Huang, and Hsin-Hsi Chen. 2021. HARGAN: heterogeneous argument attention network for persuasiveness prediction. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 13045–13054. AAAI Press.

Lu Ji, Zhongyu Wei, Jing Li, Qi Zhang, and Xuanjing Huang. 2019. Discrete argument representation learning for interactive argument pair identification. *CoRR*, abs/1911.01621.

Yohan Jo, Seojin Bang, Chris Reed, and Eduard H. Hovy. 2021. Classifying argumentative relations using logical mechanisms and argumentation schemes. *Trans. Assoc. Comput. Linguistics*, 9:721–739.

Zixuan Ke, Winston Carlile, Nishant Gurrapadi, and Vincent Ng. 2018. Learning to give feedback: Modeling attributes affecting argument persuasiveness in student essays. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial*

Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden, pages 4130–4136. ijcai.org.

Khalid Al Khatib, Henning Wachsmuth, Matthias Hagen, and Benno Stein. 2017. Patterns of argumentation strategies across topics. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 1351–1357. Association for Computational Linguistics.

Khalid Al Khatib, Henning Wachsmuth, Johannes Kiesel, Matthias Hagen, and Benno Stein. 2016. A news editorial corpus for mining argumentation strategies. In *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 3433–3443. ACL.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Tatsuki Kuribayashi, Hiroki Ouchi, Naoya Inoue, Paul Reisert, Toshinori Miyoshi, Jun Suzuki, and Kentaro Inui. 2019. An empirical study of span representations in argumentation structure parsing. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4691–4698. Association for Computational Linguistics.

Anne Lauscher, Lily Ng, Courtney Napoles, and Joel R. Tetreault. 2020. Rhetoric, logic, and dialectic: Advancing theory-based argument quality assessment in natural language processing. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 4563–4574. International Committee on Computational Linguistics.

Bin Liang, Yonghao Fu, Lin Gui, Min Yang, Jiachen Du, Yulan He, and Ruifeng Xu. 2021. Target-adaptive graph for cross-target stance detection. In *WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021*, pages 3453–3464. ACM / IW3C2.

Bin Liang, Rongdi Yin, Lin Gui, Jiachen Du, and Ruifeng Xu. 2020. Jointly learning aspect-focused and inter-aspect relations with graph convolutional networks for aspect sentiment analysis. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 150–161. International Committee on 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, ArgMining@EMNLP 2018, Brussels, Belgium, November 1, 2018*, pages 11–21. Association for Computational Linguistics.

Gaku Morio and Katsuhide Fujita. 2019. Syntactic graph convolution in multi-task learning for identifying and classifying the argument component. In *13th IEEE International Conference on Semantic Computing, ICSC 2019, Newport Beach, CA, USA, January 30 - February 1, 2019*, pages 271–278. IEEE.

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, ACL 2020, Online, July 5-10, 2020*, pages 3259–3266. Association for Computational Linguistics.

Juri Opitz and Anette Frank. 2019. Dissecting content and context in argumentative relation analysis. In *Proceedings of the 6th Workshop on Argument Mining, ArgMining@ACL 2019, Florence, Italy, August 1, 2019*, pages 25–34. 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 Köpf, 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 *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8024–8035.

Peter Potash, Alexey Romanov, and Anna Rumshisky. 2017. Here's my point: Joint pointer architecture for argument mining. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 1364–1373. Association for Computational Linguistics.

Lin Qiu, Yunxuan Xiao, Yanru Qu, Hao Zhou, Lei Li, Weinan Zhang, and Yong Yu. 2019. Dynamically fused graph network for multi-hop reasoning. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 6140–6150. Association for Computational Linguistics.

Rahul Ragesh, Sundararajan Sellamanickam, Arun Iyer, Ramakrishna Bairi, and Vijay Lingam. 2021. Hetegcn: Heterogeneous graph convolutional networks for text classification. In *WSDM '21, The Fourteenth ACM International Conference on Web Search and Data Mining, Virtual Event, Israel, March 8-12, 2021*, pages 860–868. ACM.

Wei Song, Ziyao Song, Lizhen Liu, and Ruiji Fu. 2020. Hierarchical multi-task learning for organization evaluation of argumentative student essays. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pages 3875–3881. ijcai.org.

Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958.

Christian Stab and Iryna Gurevych. 2014. Annotating argument components and relations in persuasive essays. In *COLING 2014, 25th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, August 23-29, 2014, Dublin, Ireland*, pages 1501–1510. ACL.

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

Reid Swanson, Brian Ecker, and Marilyn A. Walker. 2015. Argument mining: Extracting arguments from online dialogue. In *Proceedings of the SIGDIAL 2015 Conference, The 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue, 2-4 September 2015, Prague, Czech Republic*, pages 217–226. The Association for Computer Linguistics.

Ming Tu, Guangtao Wang, Jing Huang, Yun Tang, Xiaodong He, and Bowen Zhou. 2019. Multi-hop reading comprehension across multiple documents by reasoning over heterogeneous graphs. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 2704–2713. Association for Computational Linguistics.

Henning Wachsmuth, Khalid Al Khatib, and Benno Stein. 2016. Using argument mining to assess the argumentation quality of essays. In *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 1680–1691. ACL.

Henning Wachsmuth, Nona Naderi, Yufang Hou, Yonatan Bilu, Vinodkumar Prabhakaran, Tim Alberdingk Thijm, Graeme Hirst, and Benno Stein. 2017. Computational argumentation quality assessment in natural language. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 1: Long Papers*, pages 176–187. Association for Computational Linguistics.

Jiacheng Xu, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. Discourse-aware neural extractive text summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 5021–5031. Association for Computational Linguistics.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. Graph convolutional networks for text classification. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 7370–7377. AAAI Press.

Michihiro Yasunaga, Rui Zhang, Kshitijh Meelu, Ayush Pareek, Krishnan Srinivasan, and Dragomir R. Radev. 2017. Graph-based neural multi-document summarization. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017), Vancouver, Canada, August 3-4, 2017*, pages 452–462. Association for Computational Linguistics.

Appendices

A Different Weights for Loss Functions

Weight		F_1	
$\mathcal{L}_{am}$	$\mathcal{L}_{ape}$	AM	APE
0.25	0.75	70.01	33.98
0.5	0.5	**71.13**	**34.40**
0.75	0.25	70.51	34.33

Table 5: The results of different weights for loss functions on RR-submission (%). The best scores are in bold.

As shown in Table 5, the impacts of the different weights are minimal. The performance of the model is optimal when two weights are the same.