Semantics-Aware Dual Graph Convolutional Networks for Argument Pair Extraction

Minzhao Guan[1†], Zhixun Qiu[1†], Fenghuan Li[2*], Yun Xue[1*]
[1]School of Electronics and Information Engineering, South China Normal University,
[2]School of Computer Science and Technology, Guangdong University of Technology.
{guanminzhao, qiuzhixun, xueyun}@m.scnu.edu.cn
fhli20180910@gdut.edu.cn

Abstract

Argument pair extraction (APE) is a task that aims to extract interactive argument pairs from two argument passages. Generally, existing works focus on either simple argument interaction or task form conversion, instead of thorough deep-level feature exploitation of argument pairs. To address this issue, a Semantics-Aware Dual Graph Convolutional Networks (SADGCN) is proposed for APE. Specifically, the co-occurring word graph is designed to tackle the lexical and semantic relevance of arguments with a pre-trained Rouge-guided Transformer (ROT). Considering the topic relevance in argument pairs, a topic graph is constructed by the neural topic model to leverage the topic information of argument passages. The two graphs are fused via a gating mechanism, which contributes to the extraction of argument pairs. Experimental results indicate that our approach achieves the state-of-the-art performance. The performance on F1 score is significantly improved by 6.56% against the existing best alternative.

Keywords: Argument Pair Extraction, Dual Graph Convolutional Networks, Co-occurring Word Graph, Topic Graph, Argument Mining

1. Introduction

Argument pair extraction (APE) aims to extract interactive argument pairs from two separate passages, which is widely applied to intelligent debate (Bar-Haim et al., 2021; Slonim et al., 2021), writing assistance (Wambsganss et al., 2022; Wambsganss and Niklaus, 2022), essay scoring (Wang et al., 2018; Wachsmuth et al., 2017), and peer review (Hua et al., 2019; Fromm et al., 2021). Instead of solely extracting arguments from monologues, the interactions between reviewer comments and author rebuttals give rise to the advancement of APE task. APE is concerned, not just with settling the complicated argument structure, but also with capturing the interaction between arguments. Cheng et al. (2020) collected a large number of reviews and rebuttals to establish a dataset for APE. Figure 1 presents such examples, with a 'review' document containing comments from a reviewer while a 'rebuttal' involves the replies from an author. An argument pair is thereby formed based on each review and one corresponding rebuttal.

In general, an APE is further divided into two subtasks, i.e., argument extraction and argument pair identification. Research is ongoing to develop multitasking approaches in this field. Notwithstanding, the APE task remains challenging, primarily because these methods failed to deal with the hidden information of arguments. Existing works focused on either simplifying the structure of APE (Cheng et al., 2021; Bao et al., 2022) or leveraging only explicit features (Cheng et al., 2020; Bao et al., 2021b) (e.g., co-occurring words).

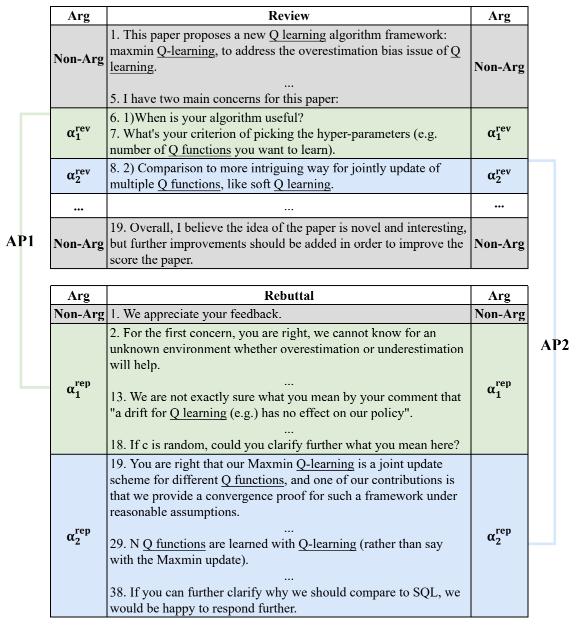


Figure 1: An example of APE. $\alpha_i^{rev}/\alpha_i^{reb}$ denotes the i^{th} argument in the Review/Rebuttal. Two distinguishing argument pairs are respectively colored in green and blue while the grey area represents non-arguments.

Encouragingly, the investigation of basic characteristics of arguments paves the way for a deeper-level analysis.

(1) As long as two arguments relate to the same issue during discussion, certain words inevitably

[†]Equal Contribution
[*]Corresponding author

LREC-COLING 2024, pages 14652–14663
20-25 May, 2024.

occur in both argument passages. The exploiting of co-occurring words substantially contributes to the argument pairing. Following the idea of (Bao et al., 2021b), the information from co-occurring words is applied to graph construction for encoding. However, the incorrect argument pairing can be generated by solely considering the co-occurring word number. For this reason, the semantic relevance of each argument has to be taken into account in addition to the number of co-occurring words.

(2) An argument pair is subject to specific topics under discussion, which can be used for argument pairing. Despite the distinctiveness of co-occurring words, the misunderstanding of argument information is also derived. According to Figure 1, the underlined word 'Q learning' is presented in the 'review' passage, as well as in the 'rebuttal', but arguments fail to form an argument pair. The application of only co-occurring words has deficiencies in argument pairing, as it is the case. Further, the passages in green and blue separately relate to the topics of parameters and algorithm comparison about 'Q learning'. As a result, the utilization of topic information benefits the argument pairing.

In this work, we propose a Semantics-Aware Dual Graph Convolutional Networks (SADGCN) that employs both co-occurring words and topic information on the task of APE. Our contributions are as follows:

- We devise a Rouge-guided Co-occurring word Graph Convolutional Network (RCGCN), which deals with the semantic relevance of argument pairs containing co-occurring words.

- A topic-related graph is constructed in line with the topic probability distribution of neural topic model and topic embeddings, which is further encoded by GCN for argument topic interaction. Lastly, a gating unit is performed to fuse the co-occurring word information and topic information for APE.

- Experiments are carried out on publicly available benchmarks to evaluate the working performance of our model. Experimental results indicate that the proposed model outperforms the state-of-the-art(SOTA) by 6.56% in F1 score. More tests are also conducted to validate the effectiveness of components in the proposed model.

The subsequent sections of this paper are organized as follows. Section 2 provides the literature overview concerning APE and the utilization of topic information in this task. In Section 3, we elucidate the definition and the form of the argument pair extraction task, aiming to enhance the comprehension of our proposed approach. Section 4 delves into the technical underpinnings of our methodology. The experimental setup for our method, along with the corresponding results, is expounded upon in Section 5. Section 6 summarizes the findings and contributions of our research. Finally, Section 7 acknowledges the limitations of our study.

2. Related Works

Recently, Cheng et al. (2020) proposed a challenging task, namely APE. This allows for a more detailed analysis that utilizes more of the information provided by both reviews and rebuttals. With respect to APE task, Cheng et al. (2020) proposed a novel multi-task model that extracts arguments from two documents using LSTM-CRF and pairs them using a classifier, which did not achieve satisfying results. In order to capture the argument pairs, Bao et al. (2021b) explicitly modeled the relation between argument pairs using co-occurrence information. While co-occurring words contributes to the argument pairing, solely considering the co-occurring word number can also lead to incorrect pairing. Therefore, we further introduced semantic information to reduce false connections. In addition, Cheng et al. (2021) transformed the APE task into an attention-guided table-filling. Besides, Bao et al. (2022) devised a two-stage machine reading comprehension (MRC) framework. However, both approaches simply transform the task form but failed to utilize the features of arguments. By contrast, we make a contribution to the mining of topic information and co-occurring word information at a deeper level instead of transforming the task mode (Cheng et al., 2021; Bao et al., 2022) and modeling the shallow-level features(Cheng et al., 2020; Bao et al., 2021b).

Topic information refers to the key issue within a text. In NLP domain, topic information is employed to facilitate the comprehension of textual content and structure, such as text classification(Zeng et al., 2018), sentiment analysis(Zhu et al., 2023; Wang et al., 2020), etc. In this context, the employment of topic information is also highlighted in argument mining tasks. Budán et al. (2020) decorated arguments with a set of related topics. Considering the topics of different arguments, Fromm et al. (2019) presented distinguishing models for argument classification. Utilizing thematic information for APE tasks is still limited. And existing approaches in the argumentation domain use a label-based paradigm to introduce topics without explicitly learning and incorporating deeper topic information. To delve into more profound information, we adopt neural topic models (NTM) (Srivastava and Sutton, 2017; Miao et al., 2017; Dieng et al., 2020) to extract pertinent topic information. This extracted information is then represented in the form of graphs. By leveraging

GCN, we effectively capture deep-seated topical connections between arguments.

3. Task Description

The objective of this work is to extract argument pairs from peer reviews and rebuttals. The APE task is carried out in the form of two sub-tasks, i.e., Argument Mining (AM) and Sentence Pairing (SP). Let $D^{rev} = \{s_1^{rev}, \ldots, s_n^{rev}\}$ denote an n-sentence review text and $D^{reb} = \{s_1^{reb}, \ldots, s_m^{reb}\}$ be its m-sentence rebuttal text. A review argument span set $X^{rev} = \{\alpha_1^{rev}, \alpha_2^{rev}, \ldots\}$ and a rebuttal argument span set $X^{reb} = \{\alpha_1^{reb}, \alpha_2^{reb}, \ldots\}$ are derived where α_i^{rev} and α_i^{reb} separately represent the i^{th} argument span of the review and rebuttal. Then, an argument pair set $P = \{p_1, p_2, \ldots\}$ with $p_i \in X^{rev} \times X^{reb}$, is obtained via sentence pairing. For the example in Figure 1, the review argument span set and rebuttal argument span set are $X^{rev} = \{\alpha_1^{rev}, \alpha_2^{rev}\} = \{(6,7), (8,8)\}$ and $X^{reb} = \{\alpha_1^{reb}, \alpha_2^{reb}\} = \{(2,18), (19,38)\}$, respectively. The argument pair set is given as $P = \{p_1, p_2\} = \{[\alpha_1^{rev}, \alpha_1^{reb}], [\alpha_2^{rev}, \alpha_2^{reb}]\}$. It is noteworthy that the single SP task requires the accurate labeling of the AM task as its input. On the other hand, the SP task in APE task takes the prediction of the AM task as its input.

4. Proposed Approach

We propose a SADGCN, which aims to determine the relationship between argument pairs by using co-occurring word information and topic information. Figure 2 shows the architecture of our model. More details of each model component are described as follows.

4.1. Input and Encoder

Seeing that APE is a document-level task, the Longformer is employed to extract context information. Following the idea proposed by Bao et al. (2022), we use a special token "$[AM]$" as the query for argument mining. This allows for the thorough mining of all arguments X^{rev} and X^{reb}. During AM process, the inputs are the concatenation of "$[AM]$" and the document, i.e.,

$$I^{AM} = ([s], [AM], [/s], [s], s_1, \ldots, s_{num}, [/s]) \quad (1)$$

where $[AM]$ is a special token used as the query to identify all the arguments. $[s]$ and $[/s]$ are special tokens of Longformer, s_i is the i^{th} sentence in the review/rebuttal document. The value of $num \in \{n, m\}$, where n and m represent the number of sentences in the review and rebuttal, respectively.

With argument mining, both $X^{rev} = \{\alpha_1^{rev}, \ldots\}$ and $X^{reb} = \{\alpha_1^{reb}, \ldots\}$ are obtained with $\alpha_t^{rev} = (s_{t,start}^{rev}, \ldots, s_{t,end}^{rev})$ and $\alpha_t^{reb} = (s_{t,start}^{reb}, \ldots, s_{t,end}^{reb})$. Taking each argument from X^{rev} as the query for SP, we concatenate α_t^{rev} with D^{reb}, to generate the input sequence, which is:

$$I_{rev \to reb,t}^{SP} = ([s], \alpha_t^{rev}, [/s], [s], s_1^{reb}, \ldots, s_m^{reb}, [/s]) \quad (2)$$

where $I_{rev \to reb,t}^{SP}$ represents the input of the t^{th} argument of X^{rev} querying the corresponding argument from D^{reb}. Similarly, we concatenate α_t^{reb} with D^{rev} to generate $I_{reb \to rev,t}^{SP}$. Note that we take the same Longformer in SP with that in AM.

Afterwards, the input sequences are fed into the Longformer model to extract the hidden representations of individual tokens. In this process, the global attention token is designated as the query token. Subsequently, the sentence representation is obtained by pooling the hidden representations of the tokens within the sentence.

4.2. Argument Mining

As depicted in Figure 2, the input data is processed by Longformer model, in order to derive the sentence representations. Subsequently, these sentence representations are fed into a Bi-LSTM+CRF architecture to yield the final output. Mathematically, the process of attention mechanism (AM) can be expressed as follows:

$$Y^{rev/reb} = \text{CRF}(\text{Bi-LSTM}(\text{Longformer}(I^{AM}))) \quad (3)$$

$$Y^{rev/reb} = \left(y_1^{rev/reb}, y_2^{rev/reb}, \ldots, y_n^{rev/reb}\right) \quad (4)$$

where $y_i^{rev/reb}$ is the IOBES tag of the i^{th} sentence in $D^{rev/reb}$.

4.3. Sentence Pairing

At this stage, one or more arguments are derived from the argument mining task, which is taken as queries for SP to obtain the sentence representations $H_{rev \to reb,t}^{SP}$ and $H_{reb \to rev,t}^{SP}$. It is worth noting that we take the gold arguments from the benchmark dataset during training. The precise modeling of argument relation from two documents is performed based on co-occurring words and topic information. To facilitate the description, we tend to present the search for rebuttal argument in D^{reb} using review argument, i.e., $rev \to reb$. The process of $reb \to rev$ is implemented in the same manner but in a reversed direction. We take the union set of both directions as the final extraction results.

4.3.1. RCGCN

Statistics reveal that over 80% of argument pairs contain co-occurring words except for stop words. However, co-occurring words can be either simply

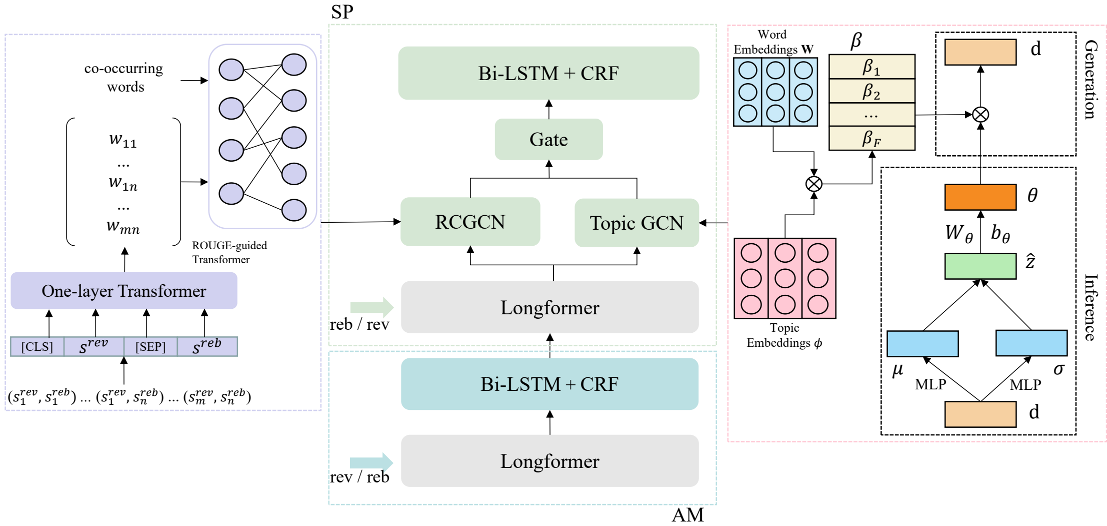


Figure 2: Model architecture.

mentioned or thoroughly discussed about them in arguments. As such, solely using the numbers of co-occurring words can lead to the unreliable pairing. In our model, the semantic relevance among words is also exploited, based on which a Rouge-guided one-layer Transformer (ROT) is established. Generally, Rouge-2 is an index for lexical relevance capture due to its capability to evaluate the lexical similarity of two texts, as it is the case in SP task. Furthermore, the one-layer Transformer is effective in dealing with semantic information (Sheng et al., 2021). In such a manner, the semantic information between lexical-related sentences is accessible and is further utilized for weight assignment in a co-occurring word-based graph.

ROT. In our model, we pre-train a Rouge-guided One-layer Transformer (ROT) to explore the semantic relevance between lexical-related sentences. Specifically, each sentence of D^{rev} is paired with each sentence of D^{reb} to obtain $n \times m$ sentence pairs. Subsequently, the Rouge-2 value of the two sentences in a pair is computed. The Transformer is initialized with the first layer of BERT. A sentence pair is fed into the Transformer. We thus have:

$$\mathrm{z_{rot}} = Transformer([CLS]s^{rev}[SEP]s^{reb}) \quad (5)$$

where $[CLS]$ and $[SEP]$ are the retained tokens and z_{rot} is the output representation.

The optimization of the ROT is performed by minimizing the mean-squared error between precision and recall of ROUGE-2, which is given by:

$$\widehat{R}(s^{rev}, s^{reb}) = MLP(z_{rot}([CLS])) \quad (6)$$

$$L_R = \left\| \widehat{\mathrm{R}}(s^{rev}, s^{reb}) - \mathrm{R}(s^{rev}, s^{reb}) \right\|_2^2 + \lambda_{\mathrm{R}} \|\Delta\theta\|_2^2 \quad (7)$$

where $\mathrm{R}(s^{rev}, s^{reb})$ is the precision and recall of sentences s^{rev} and s^{reb} calculated using the ROUGE-2 metric, $\widehat{\mathrm{R}}(s^{rev}, s^{reb})$ is the predicted precision and recall by our method, $z_{rot}([CLS])$ represents the output of the Transformer corresponding to token $[CLS]$; the first term of L_R is the regression loss; the second term of L_R is to preserve the semantic capabilities acquired by the pre-training of Transformer; λ_R is a control factor and $\Delta\theta$ refers to the parameter variation.

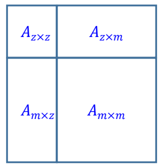


Figure 3: Components of the matrix.

Co-occurring Word Graph Construction. A graph is constructed to model the co-occurring word relation of the mined argument with the other document. Assuming that an argument α_t^{rev} contains z sentences, a lexical-relevant matrix $A^{co} \in R^{(z+m)\times(z+m)}$ can be established, where m is the number of sentences in D^{reb}. Specifically, this matrix is subdivided into four components; shown in Figure 3. The weights at the diagonal positions are set to 1 and other positions to 0 for $A^{co}_{z\times z} \in R^{z\times z}$ and $A^{co}_{m\times m} \in R^{m\times m}$. Note that $A^{co}_{z\times m} = (A^{co}_{m\times z})^T$, we shall thus focus on the construction of $A^{co}_{z\times m}$. Only if the i^{th} sentence of argument shares an occurring word with the j^{th} sentence of D^{reb}, can the edge between the two sentences be established in $A^{co}_{z\times m}$. The weight value is computed as:

$$R_i = ROT([CLS]\, s_i \,[SEP]) \quad (8)$$

$$D^{co}_{ij} = |R_i\,([CLS]) - R_j\,([CLS])\,|_1 \quad (9)$$

$$A^{co}_{ij} = 1 - \frac{D^{co}_{ij} - min(D^{co})}{max(D^{co}) - min(D^{co})} \quad (10)$$

where R_i signifies the result of i^{th} sentence obtained by ROT; R_i ($[CLS]$) and R_j ($[CLS]$) denote the vectors corresponding to token $[CLS]$ in R_i and R_j; $|\cdot|_1$ represents the number of one paradigm; $min(D^{co})$ and $max(D^{co})$ are the minimum and maximum value of D^{co}, and D^{co}_{ij} is i^{th} row and j^{th} column of D^{co}. Based on the processes above, we can construct the co-occurring word graph A^{co}.

Co-occurring Word Graph Convolution. The graph convolutional network (GCN)(Kipf and Welling, 2017) is typically used for information exchange between nodes in a graph. In our model, each node feature in the co-occurring word graph involves two parts, i.e., the sentence representation of the t^{th} argument $H^{rev}_t \in R^{z\times d}$ in D^{rev} and the sentence representation of the entire rebuttal document $H^{reb} \in R^{m\times d}$. That is, both H^{rev}_t and H^{reb} consist of the initial node features, which are presented as:

$$H^{co}_0 = [H^{rev}_t; H^{reb}] \tag{11}$$

$$H^{co}_{l+1} = \sigma([A^{co} H^{co}_l W^{co}_l]) \tag{12}$$

where $H^{co}_l \in R^{(z+m)\times d}$ contains the feature vectors of all nodes in the l^{th} layer of GCN, A^{co} is the co-occurring word adjacency matrix, W^{co}_l denotes the trainable parameter matrix and $\sigma(\cdot)$ is the ReLU activation function.

The final node features of $H^{co}_{rev\to reb}$ that are related to H^{reb} are taken as the co-occurring word graph representation $H^{co}_{reb} \in R^{m\times d}$.

4.3.2. Topic GCN

As pointed out in the section 1, the employment of topic information certainly benefits the argument pairing. In this section, we will describe the construction of the Topic Graph Convolutional Network (Topic GCN).

NTM. Neural topic model contains inference and generation. Formally, we build an inference network to infer the document-topic distribution θ. Let a bag-of-words representation $d \in R^{|V|}$ be the input to the inference network where V is the vocabulary. To start with, d is sent to two neural networks to generate $\mu(d)$ and $\sigma(d)$, together with the parameterization of $q(z|d) = N(\mu(d), \sigma^2(d))$ where $z \in R^F$ is a potential variable in the topic model, F is the topic number, $q(z|d)$ and $N(\mu(d), \sigma^2(d))$ stand for Gaussian distribution. We then re-parameterize(Kingma and Welling, 2014) the $q(z|d)$ to extract $\widehat{z} = \mu(d) + \epsilon\cdot\sigma(d)$, where ϵ is sampled from $N(0, I^2)$. The topic distribution θ is expressed as:

$$\theta = softmax(W_\theta \widehat{z} + b_\theta) \tag{13}$$

where $W_\theta \in R^{F\times F}$ and $b_\theta \in R^F$ are trainable parameter matrices.

Furthermore, the generative network is devised to parameterize $p(d|\theta, \beta)$. We define $\beta_f \in R^V$ as the word distribution of the f^{th} topic in $\beta \in R^{F\times V}$. Both the word embedding $W \in R^{V\times M}$ and the topic embedding $\Phi \in R^{F\times M}$ are used to obtain β with M representing the embedding dimension, which can be written as:

$$\beta = softmax(\frac{\Phi\cdot W^T}{\sqrt{M}}) \tag{14}$$

The reconstruction of d is presented as:

$$d = \theta\cdot\beta \tag{15}$$

The loss function of the neural topic model is given by:

$$L_{NTM} = KL[q(z|d)||p(z)] - E_{q(z|d)}[logp(d|\theta, \beta)] \tag{16}$$

where $p(z)$ represents a standard normal prior distribution $N(0, I^2)$. The first term of the loss function ensures the distribution $q(z|d)$ obtained from training approaches the real prior distribution $p(z)$. The second term represents the reconstructed document likelihood from the generative network.

Topic Graph Construction. The topic embedding of the NTM is used to construct a topic graph, which models the topic relation between the mined argument and the other document. Similar to the construction of the co-occurring word graph, for a z-sentence argument α^{rev}_t, a topic-relevant matrix $A^{topic} \in R^{(z+m)\times(z+m)}$ is established.

Specifically, the construction of $A^{topic}_{z\times m}$ starts with converting a sentence s into a bag-of-words representation d as the input to the NTM. A document-topic distribution θ is derived while the $e^{topic} \in R^M$ relates to s is computed as:

$$e^{topic} = \theta\cdot\Phi \tag{17}$$

The distance between the i^{th} sentence of α^{rev}_t and the j^{th} sentence in D^{reb} is calculated and normalized, which is:

$$D^{topic}_{ij} = |e^{topic}_i - e^{topic}_j|_2 \tag{18}$$

$$W^{topic}_{ij} = 1 - \frac{D^{topic}_{ij} - min(D^{topic})}{max(D^{topic}) - min(D^{topic})} \tag{19}$$

where D^{topic} is the distance matrix, $|\cdot|_2$ is the two-norm, W^{topic} is the weight matrix, D^{topic}_{ij} is the element of D^{topic}, as well as $max(D^{topic})$ and $min(D^{topic})$ are the maximum and minimum values in D^{topic}, respectively.

The top-k mechanism is exploited to keep the k-largest values and set other values to 0 in each row of W^{topic} and obtain the topic graph A^{topic}, where k is a hyperparameter.

$$A^{topic}_{z\times m,i} = topk(W^{topic}_i) \tag{20}$$

Topic Graph Convolution. Likewise, GCN is employed for topic information interaction between nodes.

$$H_0^{topic} = [H_t^{rev}; H^{reb}] \quad (21)$$

$$H_{l+1}^{topic} = \sigma([A^{topic} H_l^{topic} W_l^{topic}]) \quad (22)$$

where $H_l^{topic} \in R^{(z+m)\times d}$ contains the feature vectors of all nodes in the l^{th} layer of GCN, and A^{topic} is the topic adjacency matrix and W_l^{topic} denotes the trainable parameter matrix.

The final node features from $H_{rev\rightarrow reb}^{topic}$ in relation to H^{reb} are considered as the topic graph representation $H_{reb}^{topic} \in R^{m\times d}$.

4.3.3. Gated Fusion

The co-occurring word graph representation $H_{reb}^{co} \in R^{m\times d}$ and the topic graph representation $H_{reb}^{topic} \in R^{m\times d}$ are integrated via gating mechanism:

$$\alpha^{gate} = \sigma(W_1^{gate} \cdot H_{reb}^{co} + W_2^{gate} \cdot H_{reb}^{topic}) \quad (23)$$

$$H^{gate} = \alpha^{gate} \cdot H_{reb}^{co} + (1 - \alpha^{gate}) \cdot H_{reb}^{topic} \quad (24)$$

where W_1^{gate} and W_2^{gate} are trainable matrices and $\sigma(\cdot)$ stands for the sigmoid activation function.

The outcome H^{gate} is fed into LSTM to obtain its contextual representation, which is further sent to the CRF sequence tagger. The argument relates to α_t^{rev} in D^{reb} is shown as:

$$Y_{rev\rightarrow reb,t}^{pair} = (y_{rev\rightarrow reb,t1}^{pair}, \ldots, y_{rev\rightarrow reb,tm}^{pair}) \quad (25)$$

where $y_{rev\rightarrow reb,ti}^{pair}$ refers to the IOBES label for the i^{th} sentence in D^{reb}.

Based on the label sequence, one can obtain the argument span set of $X_{rev\rightarrow reb}^{reb} = \{\alpha_1^{reb}, \ldots\}$ from D^{reb} corresponding to α_t^{rev}. Notably, $X_{rev\rightarrow reb}^{reb}$ can be an empty set or contain one or multiple arguments. The argument pair set is derived as $P_{rev\rightarrow reb,t} = \{[\alpha_t^{rev}, \alpha_1^{reb}], \ldots\}$.

4.4. Model Training

The loss of our model comprises argument mining, neural topic modeling, and sentence pairing.

The loss of argument mining is computed as:

$$L_{AM} = logp(\widehat{Y}^{rev}|D^{rev}) + logp(\widehat{Y}^{reb}|D^{reb}) \quad (26)$$

where $\widehat{Y}^{rev}$ and $\widehat{Y}^{reb}$ represent the real label sequences, $p(\cdot)$ is the probability associated with CRF.

The loss L_{NTM} is described in Section 4.3.2. For the loss of sentence pairing, we have that:

$$L_{SP} = \sum_i logp(\widehat{Y}_{rev\rightarrow reb,i}^{pair}|D^{reb}, X^{rev}) + \sum_i logp(\widehat{Y}_{reb\rightarrow rev,i}^{pair}|D^{rev}, X^{reb}) \quad (27)$$

Category	Instances	Size
Review	Sentences	99.8K
	Arguments	23.2K
	Argument sentences	58.5K
	Avg. sentences per argument	2.5
Rebuttal	Sentences	94.9K
	Arguments	17.7K
	Argument sentences	67.5K
	Avg. sentences per argument	3.8
	Review-rebuttal pairs	4764

Table 1: Overall statistics of RR dataset.

where $p(\cdot)$ is the probability associated with CRF, $\widehat{Y}_{rev\rightarrow reb,i}^{pair}$ denotes the real IOBES label sequence of the i^{th} argument in D^{rev} corresponding to D^{reb}; so does $\widehat{Y}_{reb\rightarrow rev,i}^{pair}$.

Three losses are added up as the training objective of our model:

$$L = L_{AM} + L_{NTM} + L_{SP} \quad (28)$$

4.5. Inference

In the inference phase, prediction results of two reversed directions are fused for sentence pairing. For $Y_{rev\rightarrow reb,t}^{pair}$ representing the label sequence of the t^{th} argument in D^{rev} corresponding to D^{reb}, from which the rebuttal argument span can be deduced. In such a way, the argument from D^{reb} that is in pair with α_t^{rev} is extracted. The rebuttal argument span is written as $X_{rev\rightarrow reb}^{reb} = \{\alpha_1^{reb}, \ldots\}$. Accordingly, the argument pair set derived from $Y_{rev\rightarrow reb,t}^{pair}$ can be $P_{rev\rightarrow reb,t} = \{[\alpha_t^{rev}, \alpha_1^{reb}], \ldots\}$. Following this process, all argument pairs in the direction of $rev \rightarrow reb$ can be predicted as $P_{rev\rightarrow reb} = \bigcup_t P_{rev\rightarrow reb,t}$.

In the same way, all argument pairs in the direction of $reb \rightarrow rev$ can also be maintained. We shall take the predictions of both directions as the final argument pair prediction, i.e., $P = P_{rev\rightarrow reb} \bigcup P_{reb\rightarrow rev}$.

5. Experiments

5.1. Dataset

We carry out our experiment on the Review-Rebuttal(RR) dataset (Cheng et al., 2020). Details of the dataset are presented in Table 1. This dataset is of two distinguishing versions, namely RR-submission and RR-passage. In RR-submission, the review-rebuttal text pairs of the same paper are preserved in the same set. For RR-passage, the different round-review-rebuttal text pairs can be provided in different sets. The dataset is split into three subsets, namely training, validation, and test, with a ratio of 8:1:1.

Data	Method	Argument Mining			Sentence Pairing			Argument Pair Extraction		
		Pre.	Rec.	F_1	Pre.	Rec.	F_1	Pre.	Rec.	F_1
RR-submission	PL-H-LSTM-CRF	67.63	68.51	68.06	50.05	47.15	48.56	19.86	19.94	19.90
	MT-H-LSTM-CRF	70.09	70.14	70.12	53.44	42.71	47.48	26.69	26.24	26.46
	MLMC	69.53	**73.27**	71.35	60.81	47.14	53.11	37.15	29.38	32.81
	MGF	70.40	71.87	71.13	44.99	51.94	$48.22^\natural$	34.23	34.57	34.40
	MRC-APE	71.83	73.05	72.43	56.80	59.58	$58.16^\natural$	41.83	38.17	39.92
	GPT-3.5	57.83	63.31	60.45	65.64	50.57	57.13	25.02	28.57	26.68
	GPT-4	67.38	69.71	68.53	**67.33**	55.63	60.92	37.63	39.12	38.36
	Our SADGCN	**73.18**	72.88	**73.03**	59.16	**66.15**	**62.46**	**45.67**	**47.32**	**46.48**
RR-passage	PL-H-LSTM-CRF	73.10	67.65	70.27	51.34	42.08	46.25	21.24	19.30	20.22
	MT-H-LSTM-CRF	71.85	71.01	71.43	54.28	43.24	48.13	30.08	29.55	29.81
	MLMC	66.79	72.17	69.38	**61.29**	45.94	52.52	40.27	29.53	34.07
	MGF	73.62	70.88	72.22	42.45	54.00	$47.53^\natural$	38.03	35.68	36.82
	MRC-APE	**76.39**	70.62	73.39	52.22	63.11	$57.15^\natural$	37.70	44.00	40.61
	GPT-3.5	64.40	66.43	65.40	58.34	50.53	54.15	28.90	30.23	29.55
	GPT-4	68.52	71.75	70.10	59.43	56.86	58.12	38.38	40.71	39.51
	Our SADGCN	73.31	**73.69**	**73.50**	56.21	**67.37**	**61.29**	**43.25**	**47.53**	**45.29**

Table 2: Comparison results with baselines on RR-submission and RR-passage (%). The best scores are in bold. $\natural$ represents that we perform this task using their publicly available source codes.

5.2. Implementation Detail

In this experiment, a Longformer-base-4096 is used as the base encoder with an output dimension of 768. The sliding window attention size is set to 512. Besides, an AdamW optimizer is adopted. The Pytorch version is 1.11.0. The GPU is NVIDIA GeForce RTX 3090. The learning rate of Longformer is 1e-5 while that of other parameters is 1e-3. The number of GCN layers is 1. The Dropout rate is given as 0.5. The layer of LSTM is 1 with the output hidden layer dimension of 128. Our model is trained with 5 epochs and the batch size of 2. For the dataset RR-submission, the topic number F is set to 40 and the k in the top-k mechanism is 4. For RR-passage, F is set to 50 while k is 3. During the training phase, the corresponding AM golden labels of samples are input to the SP task.

5.3. Baselines

PL-H-LSTM-CRF (Cheng et al., 2020): a sequence labeling model and a sentence relationship classification model are trained independently, whose results are fused to establish argument pairs.
MT-H-LSTM-CRF (Cheng et al., 2020): Similar to PL-H-LSTM-CRF, two subtasks are trained in a multi-task framework.
MLMC (Cheng et al., 2021): an attention-guided model is developed by converting APE into a table-filling task.
MGF (Bao et al., 2021b): a GCN-based model is proposed using co-occurring word information.
MRC-APE (Bao et al., 2022): a two-stage reading comprehension framework is built.
GPT-3.5 (Ouyang et al., 2022; Brown et al., 2020): a large language model provides accurate and creative support for text generation and comprehension.
GPT-4 (Eloundou et al., 2023; OpenAI, 2023): a large-scale, multimodal model can accept image and text inputs and produce text outputs.

GPT-3.5/4 were solely prompted without undergoing fine-tuning. The baseline models, including PL-H-LSTM-CRF, MT-H-LSTM-CRF, MLMC, MGF and MRC-APE, underwent the training by means of training data.

5.4. Main Results

As shown in Table 2, the proposed model achieves the best and most consistent results on different versions of the RR dataset. Comprehensively, there is a considerable gap between the Longformer-based methods (MRC-APE and SADGCN) and Bert-based methods(MLMC and MGF). The main reason is that Longformer is capable of modeling all sentences from a document, contributing to the sentence-attentive interaction. By contrast, the sentence interaction in Bert is absent, due to its modeling one single sentence separately.

Under the condition that our model is comparable with MRC-APE in AM task, the proposed model substantially outperforms MRC-APE in APE task. The performance gaps on F1 against MRC-APE are 6.56% and 4.68% on RR-submission and RR-passage, which are significant. The exploiting of both co-occurring word and topic information shows its superiority in dealing with APE. The sentence pairing task is based on the real labels from AM. According to Table 2, with the elimination of error propagation caused by AM, our model outperforms MRC-APE by 4.30% on RR-submission and 4.14% on RR-passage of F1 score. The technical efficacy of co-occurring word and topic information in APE task is most pronounced. By contrast, the proposed model with only the number of co-occurring

Models	Argument Pair Extraction		
	Pre.	Rec.	F_1
Our SADGCN	**45.67**	**47.32**	**46.48**
W/o ROT weight	45.30	43.08	44.16
W/o RCGCN	40.65	44.03	42.27
W/o Topic GCN	45.54	40.27	42.74
W/o RCGCN & Topic GCN	36.61	38.41	37.25
W/o $D^{reb} \rightarrow D^{rev}$	45.87	41.47	43.56
W/o $D^{rev} \rightarrow D^{reb}$	41.48	39.03	40.22

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

word (MGF) fails to exceed most baselines without considering error propagation. The sole use of co-occurring word has its deficiency in dealing with APE. As such, a better working performance can be expected with the application of topic information, which is the case of our model.

Furthermore, we conducted zero-shot experiments utilizing advanced Large Language Models (LLMs) such as GPT-3.5 and GPT-4. The results of these experiments indicate that LLMs exhibit inadequate performance when it comes to addressing the task at hand. It is worth noting that LLMs demonstrate commendable performance in the subtask of Sentence Pairing (SP), effectively summarizing sentence content and deriving rationales for pairing. However, our model surpasses LLMs in terms of performance. This superiority can be attributed to several factors. Firstly, LLMs are not specifically trained for APE, which may have limited their performance in this task. Additionally, the presence of noise in the dataset could have negatively impacted the accuracy of LLMs. Lastly, further optimization of the prompt format may be necessary to enhance the performance of LLMs.

5.5. Ablation Study

In order to determine the importance of the different components in the proposed method, an ablation study is carried out; see Table 3. The most significant modules for our model are co-occurring word GCN and topic GCN, whose removals lead to drops of 4.21% and 3.74%. An even worse performance is generated by ablating these modules together. Moreover, we remove ROT by assigning the weight of 1 to edges in the co-occurring word graph, which results in a 2.32% decline of F1. Hence, the application of ROT benefits the exploiting of semantic relevance in APE.

In addition, we conducted ablation experiments to eliminate one direction. It is intuitive to assume that the APE task can be accomplished solely with the rev->reb direction. The experimental results indeed demonstrate that $rev \rightarrow reb$ ($W/o \quad D^{reb} \rightarrow D^{rev}$) is more efficient than $reb \rightarrow rev$ ($W/o \quad D^{rev} \rightarrow D^{reb}$). However, combining both directions proves to be even more effective. This is because, when tackling the argument pairing task, individuals typically identify an argument in the review paragraph and then establish argument pairs by identifying corresponding sentences in the rebuttal passage. Conversely, the process can be reversed by identifying sentences in the review paragraph that correspond to arguments derived from the rebuttal paragraph.

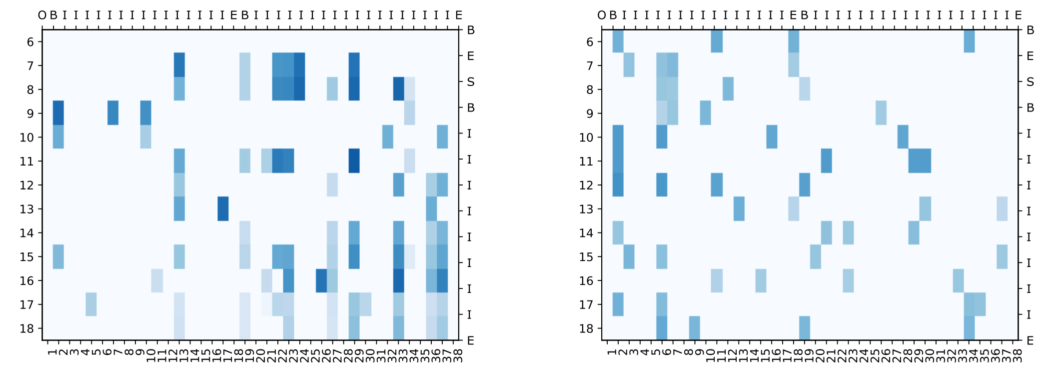
5.6. Effectiveness of Co-occurring Word Graph and Topic Graph

The example in Figure 1 is used for co-occurring word graph and topic graph visualization. In D^{rev}, the argument span set is $X^{rev} = \{\alpha_1^{rev}, \alpha_2^{rev}, \alpha_3^{rev}, \alpha_4^{rev}\} = \{(5,6),(7,7),(8,13),(14,17)\}$. Similarly, the argument span set in D^{reb} is $X^{reb} = \{\alpha_1^{reb}, \alpha_2^{reb}\} = \{(1,17),(18,37)\}$. There are four argument pairs in this example: $P = \{p_1, p_2, p_3, p_4\} = \{[\alpha_1^{rev}, \alpha_1^{reb}], [\alpha_2^{rev}, \alpha_2^{reb}], [\alpha_3^{rev}, \alpha_1^{reb}], [\alpha_4^{rev}, \alpha_2^{reb}]\}$.

In the co-occurring word graph of α_1^{rev}, sent-6 has no connection to other sentences while sent-7 relates to six sentences with only one belonging to α_1^{reb}. By contrast, both sent-6 and sent-7 in the topic graph are in relation to sentences from α_1^{reb}. One can observe that a co-occurring word 'Q functions', which is underlined in Figure 1, is captured in sent-7. However, this argument concerns 'the role of algorithms', instead of just 'Q functions'. Only based on the co-occurring word information builds the discussion about 'Q functions' and 'Q learning' in α_2^{reb}. Conversely, for α_2^{rev}, connections with sentences of α_2^{reb} are established in co-occurring word graph while those with sentences of α_1^{reb} are built in topic graph. In our model, the topic and co-occurring word are integrated with a gating mechanism to enhance the information delivery between sentences.

5.7. Execution Speed Comparison

To establish the superiority of our approach, primarily regarding its validity, we further assess its execution speed in comparison to the baseline methods. As shown in Table 4, the execution speed of LLMs tends to be suboptimal because they encompass a substantially larger quantity of parameters. PL-H-LSTM-CRF, MT-H-LSTM-CRF, MLMC and MGF exhibit a significant advantage in terms of execution speed, but their performance on F1-score is subpar. The main reason is that these models merely employ simplistic methods to solve tasks, thereby gaining an edge in execution speed. Compared to MRC models that also utilize Longformer encoding, our model engages in more interactions during the matching process. This leads to a marginally slower execution speed, but concurrently, superior performance on F1-score of our model.



(a) Co-occurring word graph. (b) Topic graph.

Figure 4: Co-occurring word graph and the topic graph. Multiple graphs of are put together for better illustration.

Dataset	Method	Execution Speed
RR-submission	PL-H-LSTM-CRF	1m12s
	MT-H-LSTM-CRF	1m03s
	MLMC	2m13s
	MGF	50s
	MRC-APE	3m21s
	GPT-3.5	53m
	GPT-4	48m
	Our SADGCN	5m25s
RR-passage	PL-H-LSTM-CRF	1m13s
	MT-H-LSTM-CRF	1m03s
	MLMC	2m14s
	MGF	52s
	MRC-APE	3m24s
	GPT-3.5	51m
	GPT-4	47m
	Our SADGCN	5m31s

Table 4: The speed of execution on test set.

6. Conclusion

In this work, a SADGCN model is developed to improve the APE task. The model incorporates co-occurring words and topic information to enhance the reliability of argument pairing. By considering the lexical and semantic relevance of arguments, the built RCGCN mitigates unreliable pairings caused by the number of co-occurring words. Additionally, a topic graph characterizes sentence relations of the same topic, enabling deeper sentence comprehension and reducing reliance on co-occurring words alone. The integration of these two types of information facilitates the extraction of argument pairs, resulting in SOTA performance on the benchmark dataset.

7. Limitations

Based on the empirical study, our model accurately extracts only 40% of arguments consisting of more than 10 sentences. This could be because AM is viewed as a task focused on annotating sentence-level sequences, making it difficult to differentiate and identify the diverse argument spans.

For another, errors generated in AM can cause the unreliability of SP results. In our work, a minor focus is to eliminate the issue of error propagation. Comparing the working performances of AM and SP, error propagation results in performance degradation of at least 15%. In general, our model is less effective in dealing with arguments of complicated sentences. Besides, the mitigation of error propagation is still in suspense, which can be addressed in future work.

8. Ethical Considerations

Our study employed the publicly available dataset, as presented by Cheng et al. (2020), for the purpose of acquiring, training, and evaluating the models. Our methodology exclusively relied on textual content and deliberately excluded the incorporation of any user profile information. It is crucial to emphasize our firm condemnation of any form of misuse of our model that may undermine data security, compromise privacy protection, or violate ethical standards.

9. Acknowledgements

We thank all anonymous reviewers for their constructive comments and useful advice. This work was supported by Natural Science Foundation of Guangdong Province (No. 2021A1515012290), Guangdong Basic and Applied Basic Research Foundation (No. 2023A1515011370), National Natural Science Foundation of China (No. 32371114), and Characteristic Innovation Projects of Guangdong Colleges and Universities (No. 2018KTSCX049).

10. Bibliographical References

Khalid Al Khatib, Lukas Trautner, Henning Wachsmuth, Yufang Hou, and Benno Stein. 2021. Employing argumentation knowledge graphs for neural argument generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4744–4754.

Khalid Al Khatib, Henning Wachsmuth, Kevin Lang, Jakob Herpel, Matthias Hagen, and Benno Stein. 2018. Modeling deliberative argumentation strategies on wikipedia. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2545–2555.

Jianzhu Bao, Chuang Fan, Jipeng Wu, Yixue Dang, Jiachen Du, and Ruifeng Xu. 2021a. 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 (Volume 1: Long Papers)*, pages 6354–6364.

Jianzhu Bao, Bin Liang, Jingyi Sun, Yice Zhang, Min Yang, and Ruifeng Xu. 2021b. Argument pair extraction with mutual guidance and intersentence relation graph. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3923–3934.

Jianzhu Bao, Jingyi Sun, Qinglin Zhu, and Ruifeng Xu. 2022. Have my arguments been replied to? argument pair extraction as machine reading comprehension. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 29–35.

Roy Bar-Haim, Liat Ein Dor, Matan Orbach, Elad Venezian, and Noam Slonim. 2021. Advances in debating technologies: Building ai that can debate humans. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: Tutorial Abstracts*, pages 1–5.

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

Maximiliano CD Budán, Maria Laura Cobo, Diego C Martinez, and Guillermo R Simari. 2020. Proximity semantics for topic-based abstract argumentation. *Information Sciences*, 508:135–153.

Liying Cheng, Lidong Bing, Ruidan He, Qian Yu, Yan Zhang, and Luo Si. 2022. Iam: A comprehensive and large-scale dataset for integrated argument mining tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2277–2287.

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)*, pages 7000–7011.

Liying Cheng, Tianyu Wu, Lidong Bing, and Luo Si. 2021. Argument pair extraction via attention-guided multi-layer multi-cross encoding. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6341–6353.

Adji B Dieng, Francisco JR Ruiz, and David M Blei. 2020. Topic modeling in embedding spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453.

Tyna Eloundou, Sam Manning, Pamela Mishkin, and Daniel Rock. 2023. Gpts are gpts: An early look at the labor market impact potential of large language models. *arXiv preprint arXiv:2303.10130*.

Michael Fromm, Evgeniy Faerman, Max Berrendorf, Siddharth Bhargava, Ruoxia Qi, Yao Zhang, Lukas Dennert, Sophia Selle, Yang Mao, and Thomas Seidl. 2021. Argument mining driven analysis of peer-reviews. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 4758–4766.

Michael Fromm, Evgeniy Faerman, and Thomas Seidl. 2019. Tacam: topic and context aware argument mining. In *IEEE/WIC/ACM International Conference on Web Intelligence*, pages 99–106.

Ivan Habernal and Iryna Gurevych. 2016. Which argument is more convincing? analyzing and predicting convincingness of web arguments using bidirectional lstm. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1589–1599.

X Hua and L Wang. 2019. Sentence-level content planning and style specification for neural text generation. In *Conference on Empirical Methods in Natural Language Processing*.

X Hua and L Wang. 2022. Efficient argument structure extraction with transfer learning and active learning. *Findings of the Association for Computational Linguistics: ACL 2022*.

Xinyu Hua, Mitko Nikolov, Nikhil Badugu, and Lu Wang. 2019. 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, Volume 1 (Long and Short Papers)*, pages 2131–2137.

Anthony Hunter and Sylwia Polberg. 2017. Empirical methods for modelling persuadees in dialogical argumentation. In *2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 382–389. IEEE.

Lu Ji, Zhongyu Wei, Jing Li, Qi Zhang, and Xuan-Jing Huang. 2021. Discrete argument representation learning for interactive argument pair identification. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5467–5478.

Yohan Jo, Jacky Visser, Chris Reed, and Eduard Hovy. 2019. A cascade model for proposition extraction in argumentation. In *Proceedings of the 6th Workshop on Argument Mining*, pages 11–24. Association for Computational Linguistics.

Diederik P Kingma and Max Welling. 2014. Auto-encoding variational bayes. *In Proceedings of ICLR-2014*.

Thomas N Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations*.

Jialu Li, Esin Durmus, and Claire Cardie. 2020. Exploring the role of argument structure in online debate persuasion. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8905–8912.

Yishu Miao, Edward Grefenstette, and Phil Blunsom. 2017. Discovering discrete latent topics with neural variational inference. In *International Conference on Machine Learning*, pages 2410–2419. PMLR.

Riki Murakami and Basabi Chakraborty. 2022. Investigating the efficient use of word embedding with neural-topic models for interpretable topics from short texts. *Sensors*, 22(3):852.

R OpenAI. 2023. Gpt-4 technical report. *arXiv*, pages 2303–08774.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Benjamin Schiller, Johannes Daxenberger, and Iryna Gurevych. 2021. Aspect-controlled neural argument generation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 380–396.

Qiang Sheng, Juan Cao, Xueyao Zhang, Xirong Li, and Lei Zhong. 2021. Article reranking by memory-enhanced key sentence matching for detecting previously fact-checked claims. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5468–5481.

Noam Slonim, Yonatan Bilu, Carlos Alzate, Roy Bar-Haim, Ben Bogin, Francesca Bonin, Leshem Choshen, Edo Cohen-Karlik, Lena Dankin, Lilach Edelstein, et al. 2021. An autonomous debating system. *Nature*, 591(7850):379–384.

Wei Song, Ziyao Song, Ruiji Fu, Lizhen Liu, Miaomiao Cheng, and Ting Liu. 2020. Discourse self-attention for discourse element identification in argumentative student essays. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2820–2830.

Akash Srivastava and Charles Sutton. 2017. Autoencoding variational inference for topic models. In *5th International Conference on Learning Representations*.

Tham Vo. 2022. An integrated topic modelling and graph neural network for improving cross-lingual text classification. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 22(1):1–18.

Henning Wachsmuth, Nona Naderi, Ivan Habernal, Yufang Hou, Graeme Hirst, Iryna Gurevych, and Benno Stein. 2017. Argumentation quality assessment: Theory vs. practice. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 250–255.

Thiemo Wambsganss, Andreas Janson, Tanja Käser, and Jan Marco Leimeister. 2022. Improving students argumentation learning with adaptive self-evaluation nudging. *Proceedings of the ACM on Human-Computer Interaction*, 6(CSCW2):1–31.

Thiemo Wambsganss and Christina Niklaus. 2022. Modeling persuasive discourse to adaptively support students' argumentative writing. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8748–8760.

Jiancheng Wang, Jingjing Wang, Changlong Sun, Shoushan Li, Xiaozhong Liu, Luo Si, Min Zhang, and Guodong Zhou. 2020. Sentiment classification in customer service dialogue with topic-aware multi-task learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9177–9184.

Yucheng Wang, Zhongyu Wei, Yaqian Zhou, and Xuan-Jing Huang. 2018. Automatic essay scoring incorporating rating schema via reinforcement learning. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 791–797.

Jian Yuan, Zhongyu Wei, Donghua Zhao, Qi Zhang, and Changjian Jiang. 2021. Leveraging argumentation knowledge graph for interactive argument pair identification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2310–2319.

Jichuan Zeng, Jing Li, Yan Song, Cuiyun Gao, Michael R Lyu, and Irwin King. 2018. Topic memory networks for short text classification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3120–3131.

Yicheng Zhu, Yiqiao Qiu, Qingyuan Wu, Fu Lee Wang, and Yanghui Rao. 2023. Topic driven adaptive network for cross-domain sentiment classification. *Information Processing & Management*, 60(2):103230.