A Simple Contrastive Learning Framework for Interactive Argument Pair Identification via Argument-Context Extraction

Lida Shi[1], **Fausto Giunchiglia**[1,2,3], **Rui Song**[1], **Daqian Shi**[3], **Tongtong Liu**[2],

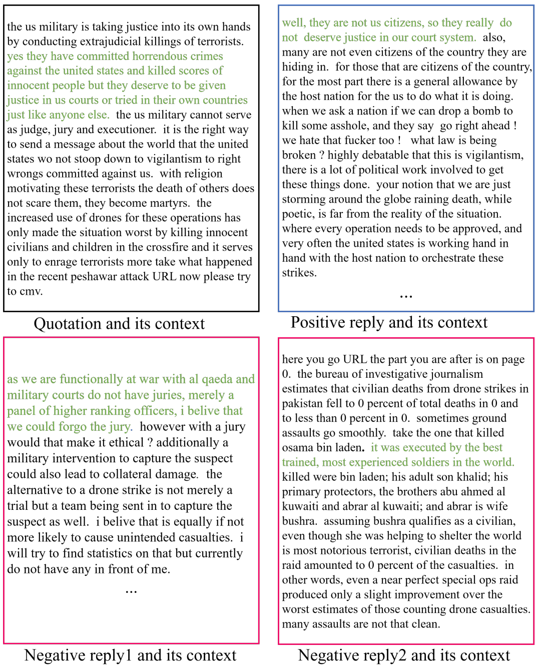
Xiaolei Diao[3], **Hao Xu**[1,2,*]
[1]School of Artificial Intelligence, Jilin University
[2]College of Computer Science and Technology, Jilin University
[3]DISI, University of Trento
{shild21,songrui20,liutt20}@mails.jlu.edu.cn, xuhao@jlu.edu.cn
{fausto.giunchiglia,daqian.shi,xiaolei.diao}@unitn.it

Abstract

Interactive argument pair identification is an emerging research task for argument mining, aiming to identify whether two arguments are interactively related. It is pointed out that the context of the argument is essential to improve identification performance. However, current context-based methods achieve limited improvements since the entire context typically contains much irrelevant information. In this paper, we propose a simple contrastive learning framework to solve this problem by extracting valuable information from the context. This framework can construct hard argument-context samples and obtain a robust and uniform representation by introducing contrastive learning. We also propose an argument-context extraction module to enhance information extraction by discarding irrelevant blocks. The experimental results show that our method achieves the state-of-the-art performance on the benchmark dataset. Further analysis demonstrates the effectiveness of our proposed modules and visually displays more compact semantic representations. The code is available at GitHub [1].

1 Introduction

Computational argumentation, as a branch of natural language understanding, has become a new research field. Existing work can be divided into two categories (Asterhan and Schwarz, 2007): monological argumentation and dialogical argumentation. Monological argumentation is the scenario for one participant, such as RCT (Mayer et al., 2020), student essays (Stab and Gurevych, 2014) and user comments (Niculae et al., 2017). The researchers focus on topics like argumentation (argument) mining (Galassi et al., 2018; Morio et al., 2020; Jo et al., 2019; Ruiz-Dolz et al., 2021), argument assessment (Anne et al., 2020; Skitalinskaya et al., 2021), and argument reasoning (Botschen et al., 2018; Habernal et al., 2018; Ruiz-Dolz et al., 2021) for this sort of study. Recently, researchers have been paying great attention to dialogical argumentation, since online forums have become the primary medium for argumentation and discussion.



the us military is taking justice into its own hands by conducting extrajudicial killings of terrorists. yes they have committed horrendous crimes against the united states and killed scores of innocent people but they deserve to be given justice in us courts or tried in their own countries just like anyone else. the us military cannot serve as judge, jury and executioner. it is the right way to send a message about the world that the united states wo not stoop down to vigilantism to right wrongs committed against us. with religion motivating these terrorists the death of others does not scare them, they become martyrs. the increased use of drones for these operations has only made the situation worst by killing innocent civilians and children in the crossfire and it serves only to enrage terrorists more take what happened in the recent peshawar attack URL now please try to cmv.

Quotation and its context

well, they are not us citizens, so they really do not deserve justice in our court system. also, many are not even citizens of the country they are hiding in. for those that are citizens of the country, for the most part there is a general allowance by the host nation for the us to do what it is doing. when we ask a nation if we can drop a bomb to kill some asshole, and they say go right ahead ! we hate that fucker too ! what law is being broken ? highly debatable that this is vigilantism, there is a lot of political work involved to get these things done. your notion that we are just storming around the globe raining death, while poetic, is far from the reality of the situation. where every operation needs to be approved, and very often the united states is working hand in hand with the host nation to orchestrate these strikes.

...

Positive reply and its context

as we are functionally at war with al qaeda and military courts do not have juries, merely a panel of higher ranking officers, i belive that we could forgo the jury. however with a jury would that make it ethical ? additionally a military intervention to capture the suspect could also lead to collateral damage. the alternative to a drone strike is not merely a trial but a team being sent in to capture the suspect as well. i belive that is equally if not more likely to cause unintended casualties. i will try to find statistics on that but currently do not have any in front of me.

...

Negative reply1 and its context

here you go URL the part you are after is on page 0. the bureau of investigative journalism estimates that civilian deaths from drone strikes in pakistan fell to 0 percent of total deaths in 0 and to less than 0 percent in 0. sometimes ground assaults go smoothly. take the one that killed osama bin laden. it was executed by the best trained, most experienced soldiers in the world. killed were bin laden; his adult son khalid; his primary protectors, the brothers abu ahmed al kuwaiti and abrar al kuwaiti; and abrar is wife bushra. assuming bushra qualifies as a civilian, even though she was helping to shelter the world is most notorious terrorist, civilian deaths in the raid amounted to 0 percent of the casualties. in other words, even a near perfect special ops raid produced only a slight improvement over the worst estimates of those counting drone casualties. many assaults are not that clean.

Negative reply2 and its context

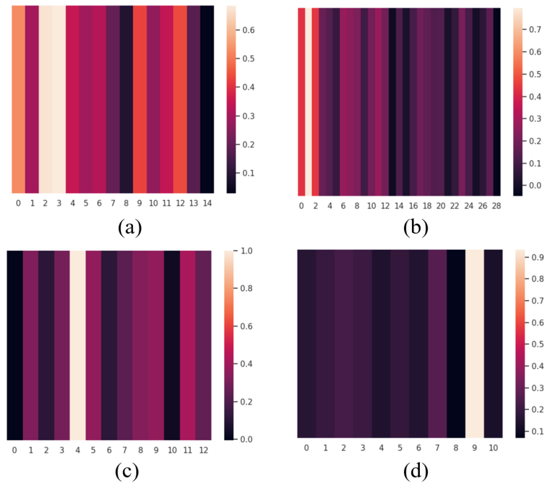
Figure 1: An instance in the dataset. Each instance includes six arguments: a quotation and its corresponding five candidate replies. Additionally, the task provides contextual information for each argument. For this task, the model needs to identify whether the quotation and the reply are interactively related. Only one of the five candidate replies is correct. The arguments are represented in green font, and the context is expressed in black.

People can express themselves on the network

[*] Corresponding author

[1]https://github.com/shilida/CL_Interactive_Argument_Pair_Identification

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 10027–10039
December 7-11, 2022 ©2022 Association for Computational Linguistics



(a) (b) (c) (d)

Figure 2: Four heatmaps of the semantic similarity between the argument and each sentence in its context. The horizontal coordinate is the number of sentences, and the vertical coordinate is the range of semantic similarity. (a) quotation-context (b) positive relpy-context (c) negative relpy1-context (d) negative relpy2-context.

anywhere and at any time, thanks to the widespread use of the Internet and communication technologies. Indeed, different people have diverse arguments on a subject, and argumentation is the most effective way to interchange arguments. Many online forums, such as *ChangemyView*[2] and *idebate*[3], provide a venue for free online argumentation, allowing users to argue with others regardless of time or location. Therefore, the study of argumentation in the interactive text arises. Earlier research (Wei et al., 2016; Tan et al., 2016) uses the data from the *ChangemyView* forum to focus on the key elements of persuasion arguments. Then, (Lu et al., 2021) formulates an interesting and meaningful task to identify whether two arguments are interactively related. More interestingly, (Yuan et al., 2021a) have applied the task to the legal field to help the court to pinpoint the focus of the case by analyzing the arguments of both sides in the trial transcript and allow the judge to make a fair decision. Figure 1 demonstrates the details of this task.

Obviously, it is difficult to identify the interactive relationship by two arguments because most arguments contain only a few words. Moreover, contextual information is related to the meaning of the quotation and reply. Thus, it is essential to utilize contexts. (Lu et al., 2021) propose a hierarchical RNN network to model context. (Yuan et al., 2021b) constructs the argumentation knowledge graph to extract entity information from the context. However, the current context-based methods achieve limited improvement since the entire context normally contains a large amount of irrelevant information. Figure 2 shows the heatmaps of the semantic similarity between the argument and each sentence in its context for the instance of Figure 1. Remarkably, many sentences in the context have very low semantic similarity to the argument, and some are even close to 0. Intuitively, as shown in Figure 1, the quotation talks about "terrorists deserve justice in court" while its context mentions "religion", "drones", and other irrelevant information. "drones" can also be found in the contexts of the two negative replies. If the whole context is modeled, the model is likely to infer the interactive relationship between quotation and negative reply2. Undoubtedly, these irrelevant sentences are noisy data for this task, negatively affecting the model training.

In this paper, we propose a simple contrastive learning framework to enhance the robustness of the model under noise conditions and reduce the adverse effects on model. This framework can construct hard argument-context samples by randomly extracting the context blocks. We combine the cross-entropy and the supervised contrastive loss to improve the expressiveness of the representations. In addition, we propose an argument-context extraction (ACE) module to enhance context information extraction. In this module, we can obtain the semantic similarity of argument and context blocks and further extract the context blocks with high similarity as the model's input. Through empirical analysis, we observe that our model performs better with the benchmark dataset and noisy dataset, which proves the superiority of our method. Our main contributions can be summarized as follows:

- We propose a simple contrastive learning framework to obtain robust and uniform semantic representation.

- We design an argument-context extraction (ACE) module to enhance information extraction by discarding irrelevant blocks.

- The experimental results show that our method achieves the state-of-the-art performance on the benchmark dataset. Further analysis demonstrates the effectiveness of our

[2] https://www.reddit.com/r/changemyview/
[3] https://idebate.org/

proposed modules and visually displays more compact semantic representations.

2 Related Work

2.1 Argumentation Mining

Argumentation (argument) mining aims to identify writing structures (such as claims, evidence, and statements) and detect the existing relations from the texts (Lytos et al., 2019; Lawrence and Reed, 2020). A lot of methods have been proposed in previous studies such as BiLstm (Eger et al., 2017), multi-task learning (Galassi et al., 2021, 2018), attentive residual networks (Galassi et al., 2021), unsupervised knowledge (Dutta et al., 2022), transformer-based model (Ruiz-Dolz et al., 2021; Mayer et al., 2020), and cascade model (Jo et al., 2019). In addition, many researchers have applied the task to many scenarios such as healthcare (Mayer et al., 2020), education (Stab and Gurevych, 2014; Alhindi and Ghosh, 2021), peer reviews(Niculae et al., 2017).

Different from monological argumentation mentioned above, an increasing number of academics begin to conduct studies on dialogical argumentation. (Ji et al., 2018) investigates the issue of persuasiveness evaluation for argumentative comments. (Cheng et al., 2020) introduces a new argument pair extraction task on peer review and rebuttal to study the contents, structures and connections between them. Similarly, (Lu et al., 2021) propose the task of identifying the interactive argument pair in online debate forum. Subsequently, (Yuan et al., 2021b) leverages a knowledge graph (Khatib et al., 2020) to model the contextual information and encodes the entity and path in the context to obtain entity embedding and path representation.

2.2 Contrastive Learning in NLP

Contrastive learning (CL) has gained tremendous attention in the natural language processing (NLP) field. The main idea is to train a representation layer by pulling closer representations of the positive samples and separating them from negative ones. Contrastive learning can be divided into self-supervised contrastive learning and supervised contrastive learning. Positive and negative samples have different definitions in different scenarios. In self-supervised contrastive learning, (Fang et al., 2020) propose a pre-trained language representation model (CERT) using contrastive learning at the sentence level to facilitate the language understanding tasks. (Gao et al., 2021) propose a simple sample augmentation strategy by just adjusting dropout masks in contrastive learning framework and advances the state-of-the-art sentence embeddings. In supervised contrastive learning, (Gao et al., 2021) incorporates annotated pairs from natural language inference datasets into the contrastive learning framework, by using "entailment" pairs as positives and "contradiction" pairs as hard negatives. Inspired by (Khosla et al., 2020), (Gunel et al., 2020) propose a new supervised contrastive loss(SCL). Combined with cross-entropy, the new SCL loss obtains significant improvements on multiple datasets of the GLUE benchmark in few-shot learning settings.

3 Method

3.1 Task Definition

Figure 1 demonstrates the details of this task. This task contains two kinds of arguments: quotation and reply. For a quotation q and its context c_q, it has five candidate replies $\{r_i\}_{i=1}^5$ with their responding contexts $\{c_{r_i}\}_{i=1}^5$. The model needs to identify whether the quotation and the reply are interactively related. Only one of the five candidate replies is correct. arg is the general term for q and r. Previous research (Yuan et al., 2021b; Lu et al., 2021) treats the task as a sentence pair ranking problem. In this paper, we treat the task as a binary classification problem. If two arguments are interactively related, the label is 1. Otherwise, the label is 0.

3.2 Argument-context Extraction Module

In this paper, we introduce the idea of information retrieval to discard irrelevant information in the context. Inspired by (Li and Gaussier, 2021; Li et al., 2021), the argument-context extraction module is based on three main steps: (1) Context block segmentation (2) Argument-context similarity calculation (3) Context block selection. The structure is shown in the Figure 3. The following describes each step in detail.

3.2.1 Context Block Segmentation

Here, we adopt the dynamic programming method (Ding et al., 2020) to segment context into blocks. The main idea of the method is to segment a document into multiple blocks by punctuation, and the block size is a hyperparameter (denoted as α in this paper). It sets different costs for different

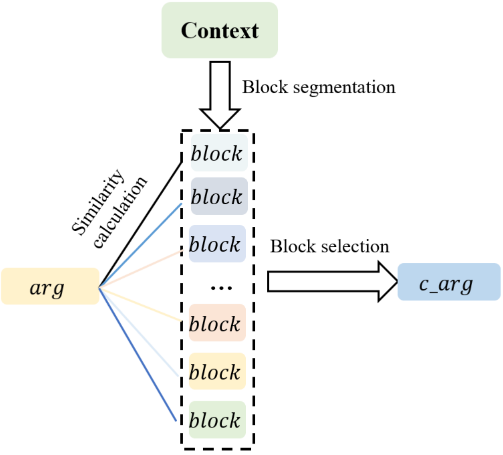


Figure 3: An illustration of argument-context extraction module.

punctuation marks to segment in priority on strong punctuation marks such as ".", "?" and "!". This process may damage the coherence of the whole context, but we consider that some redundant context blocks will be detrimental to classification. In other words, a few key blocks in the context store sufficient and necessary information to fulfill this task, which is why the redundant context should be removed. The algorithm are showed in Appendix A.

3.2.2 Argument-context Similarity Calculation

After the block segmentation module, c_{arg} is segmented into N blocks. Next, we evaluate the semantic relevance between each block and arg by calculating the cosine similarity of its embedding. The equation are showed as follows:

$$Sim\left(arg, c_{arg}\right) = \begin{bmatrix} sim\left(h_{arg}, h_{blcok1}\right) \\ sim\left(h_{arg}, h_{blcok2}\right) \\ \cdots \\ sim\left(h_{arg}, h_{blcokN-1}\right) \\ sim\left(h_{arg}, h_{blcokN}\right) \end{bmatrix} \tag{1}$$

where $h = BERT_{\theta}(x)$ is the sentence embedding. In this work, we use the BERT pre-trained by (Gao et al., 2021) for encoding sentences into embeddings. $sim\left(h_1, h_2\right)$ is the cosine similarity $\frac{h_1^T h_2}{\|h_1\|\|h_2\|}$. $Sim\left(arg, c_{arg}\right)$ is the similarity vector between arg and the context block.

3.2.3 Context Block Extraction

For this task, we propose a new input form of BERT to combine arg and its context. According to the above steps, the most relevant context blocks to arg is obtained by ranking the $Sim\left(arg, c_{arg}\right)$. Next, the most relevant blocks are concatenated together (in their order of appearance in the context) and with the arg. Finally, we use $[SEP]$ to separate the quotation part from the reply part. The equation are showed as follows:

$$c_q^b = c_q^{b1}, c_q^{b2}, \ldots c_q^{bn} \tag{2}$$

$$c_r^b = c_r^{b1}, c_r^{b2}, \ldots c_r^{bn} \tag{3}$$

$$z = [CLS]\, q, c_q^b\, [SEP]\, r, c_r^b\, [SEP] \tag{4}$$

where z is the input of the BERT. c_q^b is the top $n\,(n \leq N)$ blocks that are most similar to q. Similarly, c_r^b is the top n blocks that are most similar to r. Note that the number n of selected blocks depends on the capacity of BERT and block size α. The token length relationship is defined as follows:

$$3+L(q)+\sum_{i=1}^{n} L\left(c_q^{bi}\right)+L(r)+\sum_{i=1}^{n} L\left(c_r^{bi}\right) \leq 512 \tag{5}$$

where $L(x)$ is the length of x. To prevent information loss, we try to satisfy the above inequality when setting n and α. If the input length is longer than 512, we use hard truncation to comply with the input limit of BERT.

3.3 Contrastive Learning Framework

Prior work (Gunel et al., 2020; Gao et al., 2021) has demonstrated that contrastive learning is effective for learning sentence embedding by pulling closer representations of the positive samples and separating them from negative ones. Inspired by this, we introduce the contrastive learning objective into argument pair recognition and propose a new hard sample construction method. The detailed architecture of contrastive learning for interactive argument pair identification is shown in Figure 4.

3.3.1 Definition of Positive and Negative Samples

In self-supervised contrastive learning, positive and negative samples construction is a fascinating question. Many works (Chuang et al., 2022; Wu et al., 2021) try to find excellent methods for constructing positive and negative samples. In supervised contrastive learning, the samples are labeled so that positive and negative examples can be easily obtained. For this task, we treat the task as a binary classification problem. If two arguments are interactively related, we define them as a positive

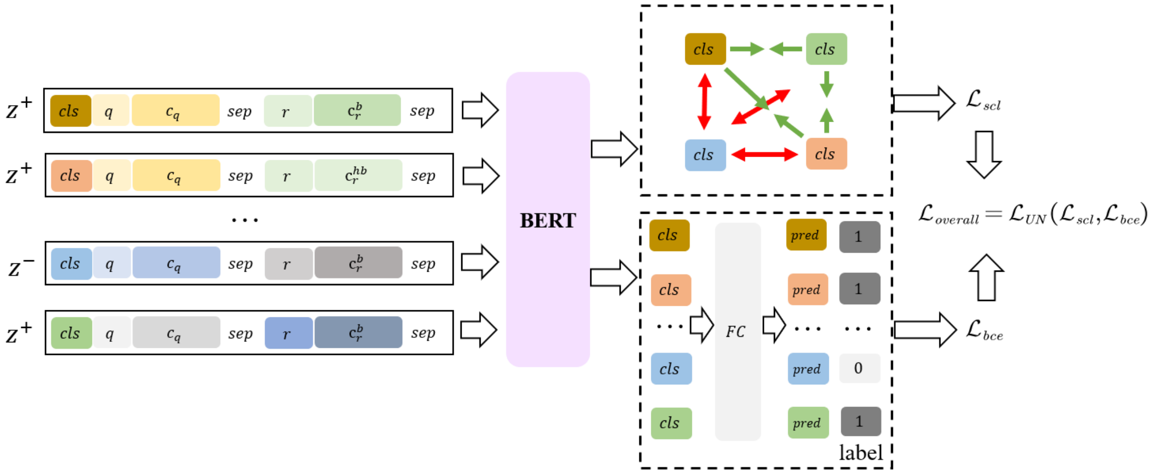


Figure 4: An illustration of our framework. Note that different colored blocks denote different values.

sample and denote it by z^+. Otherwise, we define them as a negative sample and denote it by z^-.

3.3.2 Hard Samples Construction

We propose a hard sample construction method in order to enhance the robustness of the model under noise conditions. Our method is very simple. As shown in 3.2, different block sizes and block selection rules generate different blocks in the contextual segmentation module. We use different block sizes and block selection rules to construct hard samples. Specifically, we use the strategy of randomly selecting context blocks to construct hard samples. When using the random selection strategy, more irrelevant information is introduced. It will be more difficult for the model to identity the interactive relationship between two arguments than the input using a high similarity selection strategy. The equation is as follows:

$$z_{hard} = [CLS]\, q, c_q^b\, [SEP]\, r, c_r^{hb}\, [SEP] \quad (6)$$

$$c_r^{hb} = c_r^{b1}, c_r^{b2}, ...c_r^{bm} \quad (7)$$

$$c_q^b = c_q^{b1}, c_q^{b2}, ...c_q^{bn} \quad (8)$$

where z_{hard} denotes the constructed hard sample, using a different background block size and random block selection strategy compared to the original sample. c_r^{hb} denotes the context of the reply for more irrelevant information. Note that c_r^{hb} and c_r^b are different, and $m \neq n$. In practice, for each positive sample, we construct three hard samples corresponding to it. For each negative sample, we construct one hard sample. The hard samples construction is essentially a data augmentation method from the data perspective. On the one hand, it increases the complexity and diversity of the dataset. On the other hand, it alleviates the problem of unbalanced data distribution (previously 1:4, now 1:2).

3.3.3 Training Objectives

Our framework contains two training objectives: binary classification and contrastive learning. For binary classification, we use binary cross-entropy loss. For contrastive learning, we use a supervised contrastive learning paradigm. Specifically, we introduce a supervised contrastive learning loss (Gunel et al., 2020) formulated to push representations from the same class close and representations from different classes further apart. The loss function is defined as follows:

$$\mathcal{L}_{bce} = -\frac{1}{N}\sum_{i=1}^{N} y_i \log \hat{y}_i + (1 - y_i)\log(1 - \hat{y}_i) \quad (9)$$

$$\mathcal{L}_{scl} = -\frac{1}{N}\sum_{i=1}^{N} \frac{1}{N_{y_i}-1} \sum_{j=1,i\neq j,y_i=y_j}^{N} \quad \Phi \quad (10)$$

$$\Phi(h_i, h_j) \;= \log \frac{e^{sim(h_i,h_j)/\tau}}{\sum_{k=1,k\neq i}^{N} e^{sim(h_i,h_k)/\tau}} \quad (11)$$

In $\mathcal{L}_{bce}$, y_i denotes the label of i_{th} sample and $\hat{y}_i$ denotes the model output for the probability of i_{th} sample. In $\mathcal{L}_{scl}$, N_{y_i} is the total number of samples in the mini-batch that have the same label as y_i. τ is an adjustable scalar temperature hyperparameter that controls the separation of classes.

3.3.4 Uncertainty Weighting

We introduce the Uncertainty Weighting (UW) (Kendall et al., 2018) to learn the weights between contrastive learning and binary classification. It dynamically weights multiple loss functions by considering the homoscedastic uncertainty of each objective. Therefore, it can combine losses of different orders of magnitude. Specifically, it rewrites the joint loss function as the following weighted sum:

$$\mathcal{L}_{UW}\left(\mathcal{L}_1, \mathcal{L}_2\right) = \frac{1}{2\sigma_1^2}\mathcal{L}_1 + \frac{1}{2\sigma_2^2}\mathcal{L}_2 + \log \sigma_1 \sigma_2 \tag{12}$$

where σ denotes the model's observation noise parameter to capture how much noise we have in the outputs. It is a learnabel parameter. Specifically, σ_1 and σ_2 control the relative weights of the $\mathcal{L}_1$ and $\mathcal{L}_2$, respectively. $\log \sigma_1 \sigma_2$ is a regularization term to prevent σ too large. For our task, we use uncertainty weighting to combine contrastive learning loss $\mathcal{L}_{scl}$ with binary classification loss $\mathcal{L}_{bce}$ as the overall loss:

$$\mathcal{L}_{overall} = \mathcal{L}_{UW}\left(\mathcal{L}_{bce}, \mathcal{L}_{scl}\right) \tag{13}$$

In this paper, we can adaptively adjust the two objectives during the training process by the Uncertainty Weighting.

4 Experiments

4.1 Experiment Setup

4.1.1 Experimental Dataset

The dataset[4] we use is constructed by (Lu et al., 2021). The data collection is built on the *ChangemyView* dataset (Tan et al., 2016). For this task, each instance includes the quotation, one positive reply, four negative replies and their contexts. The number of instances in training and testing set is 11565 and 1481, respectively. Similar to the previous (Lu et al., 2021; Yuan et al., 2021b), we randomly split 10% of the training set as validation set. In our experiment, the number of instances in training set and validation set is 10408 and 1157, respectively.

4.1.2 Implementation Details

The output hidden of BERT dimensions are 768. Dropout is used as 0.1 to avoid overfitting. We use AdamW (Loshchilov and Hutter, 2018) as our optimizer and the weight decay set to 1×10^{-8}. The max length of sequence is set to 512, and initial learning rate is set as 1×10^{-4}. The model is trained on the training set for 5 epochs and batch size is 40. For the normal samples, the block size and number of block are set to 6 and 42. For the hard samples, there are three alternative options. The block size and number of block are set to $4, 5, 8$ and $64, 56, 32$. We implement our code using Pytorch (Paszke et al., 2019) and Huggingface Transformers (Wolf et al., 2020) libraries. The hyperparameter τ is set as 0.03. The experiments are conducted on an NVIDIA V100 32GB GPU.

4.1.3 Models for Comparison

In order to demonstrate the effectiveness and superiority of our method, we compare with many state-of-the-art methods. The main comparison methods are as follows:

- BERT without Context (Devlin et al., 2019): This method fine-tunes the BERT for sentence pair classification. This method only utilizes the quotation and reply, and does not make use of their contextual information. The input form of BERT without context is $z = [CLS]\, q\, [SEP]\, r\, [SEP]$.

- Hierarchical Context (Lu et al., 2021): This method designs a discrete variational autoencoders (DVAE) to extract the representation of quotation and replies. A hierarchical structure is proposed to obtain the representation of the context by BiGRU. Finally, it integrates quotation or replies representations and their contextual representations to obtain the final sentence encoding.

- Knowledge Graph and GCN (Yuan et al., 2021b): This method is very sophisticated and it is the stat-of-the-art method so far. Firstly, (Yuan et al., 2021b) constructs a dialogical argumentation knowledge graph. Then, it uses a path-based graph convolutional network to encode the concepts and the reasoning path between concepts from the contexts. Finally, it aligns the conceptual information with the semantic information obtained by BERT.

4.2 Overall Performance

Previous methods (Lu et al., 2021; Yuan et al., 2021b) treat the task as a sentence pair ranking

[4] http://www.sdspeople.fudan.edu.cn/zywei/data/arg-pairs-fudanU.zip

Method	P@1(%)	MRR(%)
Random Guess	20	45.67
BiGRU	51.52	70.57
BiGRU+RNN Context	55.98	73.20
BiGRU+Hierarchical Context	57.46	73.72
VAE+Hierarchical Context	58.61	74.66
DVAE+Hierarchical Context	61.17	76.16
BERT	61.85	76.57
BERT+Hierarchical Context	66.85	78.51
BERT+Knowledge Graph+GCN+Context*	68.75	80.85
Ours	**82.17**	**89.60**

Table 1: Experimental results of our method and other former methods on the test dataset, where the sign "*" represents the state-of-the-art method.

problem. Precision at one (P@1) and mean reciprocal rank (MRR) are used as evaluation metrics. For comparison purposes, we also use them as metrics. For the calculation of MRR, we use the classification probabilities to produce ranks. The results are listed in Table 1. From the table, we can make the following observations:

- The introduction of contextual information is crucial for this task. When adding contextual information, all methods are better than those before adding contextual information. Therefore, how to make better use of the contextual information is essential for this task.

- Compared with the state-of-the-art method, our method shows an amazing performance improvement. We observe that our method outperform the state-of-the-art method by 13.42% and 8.75% in P@1 and MRR, respectively. There are two main reasons: firstly, the argument-context extraction module (Section 3.2) can select the most important context blocks for the current quotation and reply, thus reducing the interference of redundant information to the model. It reconfirms the importance of making full use of contextual information. Secondly, the introduction of contrastive learning enables the model to learn more robust semantic embeddings, substantially improving the model's ability to discriminate argument pairs. In addition, compared to the previous complicated model (Lu et al., 2021; Yuan et al., 2021b), we only use the BERT, which is extremely elegant and reduces the number of parameters.

Method	P@1(%)	MRR(%)
BERT-BCE(baseline)	63.54	77.82
+ ACE	75.35	85.34
+ CL	80.01	88.35
+ Hard	**82.17**	**89.60**

Table 2: Ablation study on each module. "BERT-BCE" denotes the BERT trained by binary cross entropy loss. "ACE" denotes the argument-context extraction module module (Section 3.2). "CL" denotes the contrastive learning (Section 3.3). "Hard" denotes the contrastive learning with hard samples construction(Section 3.3.2). The best results are highlighted in bold. The same below.

4.3 Ablation Study

In this section, we investigate the quantitative impact of each module on the final performance. The results of the ablation study are shown in the Table 2. We use the BERT trained binary cross entropy loss as the baseline. Note it does not use the contextual information(detailed in section 4.1.3). After adding the argument-context extraction module, the experimental results show a remarkable improvement in both metrics. The model's performance improves by 11.81% in P@1 and 7.52% in MRR, which directly demonstrates the effectiveness of the argument-context extraction module. In addition, it also shows that the context contains a lot of valuable information, which is essential for this task. The introduction of contrastive learning improves the performance by 4.66% in P@1 and 3.01% in MRR. Obviously, the model can learn more robust semantic representations by adding the training objective of contrastive learning. Here,

Method	P@1(%)	MRR(%)
Without context	63.54	77.82
Low similarity	69.14	81.40
Random	73.60	84.15
High similarity(ours)	**75.35**	**85.34**

Table 3: Performance comparison under different block selection strategies. "Low similarity" denotes the selection of the blocks with a low similarity ranking among the candidate blocks. "Random" denotes the random selection. "High similarity (ours)" denotes the selection of the blocks with a high similarity ranking among the candidate blocks. The best results are highlighted in bold.

we use uncertainty weighting(detailed in section 3.3.4) to blend the contrastive loss and BCE loss by default. Finally, the model performance is also significantly improved by constructing more hard samples. Note that the hard sample construction is a changeable module, although the hard sample construction can promote the effect of contrastive learning.

5 Further Analyses

5.1 Analysis on ACE Module

To further validate effectiveness of the argument-context extraction module, we explore the context block selection strategy, and the results are as shown in Table 3. We can make the following observations. Firstly, it again demonstrates the importance of contextual information and the effectiveness of the argument-context extraction module. Even using "low-similarity" blocks, our method achieves a significant performance improvement compared to baseline(5.6% in P@1). Secondly, using "low-similarity" blocks also achieves great results. We consider "low-similarity" blocks also have a lot of valuable information because some contexts only obtain a few blocks after segmentation. Many overlapping blocks in "high-similarity" and "low-similarity" blocks. Finally, compared with "Low similarity" and "Random", "High similarity" achieves significant improvement, which shows that context contains redundant information harmful to model identification.

5.2 Analysis on Hard Samples Construction

In section, we further explore the effect of the hard samples construction module and explain why it works. Further experimental results are shown in the table 4. The performance improvement can be ablated into two parts: (1) hard samples construction module without contrastive learning (2) hard samples construction module with contrastive learning. Without contrastive learning, constructing hard samples is comparable to a data augmentation strategy. The performance is also significantly improved compared to only ACE module. We consider two explanations for this phenomenon. On the one hand, adding hard samples to the original dataset increases the scale of the dataset, and thus the model achieves better performance. On the other hand, we construct many positive samples, which somewhat smooth the ratio of positive to negative samples (previously 1:4, now 1:2). With contrastive learning, the performance is further improved. It is because contrastive loss is a hardness-aware loss (Wang and Liu, 2021; Gunel et al., 2020). τ controls the strength of penalties on hard samples. Though experimental and empirical analysis (detailed in Appendix B.2), we set τ to 0.05. In this experimental condition, the model focuses more on hard samples, resulting in a more uniform representation and better performance.

Method	P@1(%)	MRR(%)
ACE	75.35	85.34
Hard without CL	80.62	88.75
Hard with CL	**82.17**	**89.60**

Table 4: Further experimental results on hard samples construction.

Method	O	(a)	(b)	(c)
BCE	80.62	74.61	80.21	78.19
BCE+CL	**82.17**	**76.10**	**81.17**	**78.53**

Table 5: Results on noisy testing sets with varying kinds of noise. "O" denotes the original text. "(a),(b),(c)" denote the three kinds of noisy. We use P@1 as the metric.

5.3 Robustness on Noisy Dataset

To evaluate the robustness and stability, we add some noise in our testing set for experiments. We design three noises: (a) select low similarity context blocks instead of high similarity (b) apply augmentation randomly (swap, crop, delete) (c) simulate keyboard distance error. An example of constructing a noisy sample is shown in the table 6. In practice, we use the NLPAUG (Ma, 2019) library. In table 5, we report our results on noisy testing set with different kinds of noise. Obviously, consistent

Noisy method	Text
Original text	i am willing to bet that john boehner would have an easier time dealing with congress as president than joe biden would due to his constant interaction with it.
Augmentation randomly	am willing that john have an easier time dealing with congress as president than joe would due his interaction it.
Simulate keyboard distance error	am !Jllijg rhaR john have an easier time vWalinb S7th dpnnress as president rhwn joe 1ouKd due his interaction it.

Table 6: An instance of constructing a noisy sample.

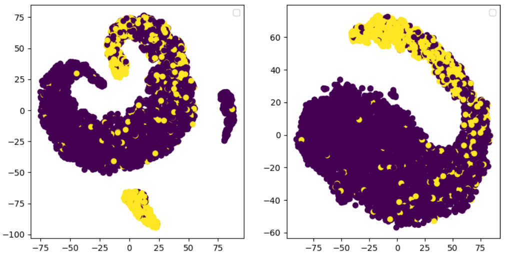


Figure 5: t-SNE plots of the learned CLS embeddings on the testing set. Left: BCE; Right: BCE+CL; Violet: negative examples; Yellow: positive examples.

improvements over the CL with BCE+CL across all noise kinds, which shows that our method leads to models that are more robust to different kinds of noise in the testing data.

5.4 Visualization

In figure 5, we show t-SNE (Van der Maaten and Hinton, 2008) plots of the learned representations of the CLS embeddings on testing set. We can clearly observe that the BCE+CL term enforces a more compact clustering of examples with the same label, while the distribution of the embeddings learned with BCE is not compact. It shows that we obtain a robust and uniform representation by introducing contrastive learning.

6 Conclusion and Future Work

This paper proposes a simple contrastive learning framework which provides a new perspective on data augmentation with text input for this task. It can be extended to other similar tasks in language model fine-tuning. Besides, we propose an argument-context extraction (ACE) module to enhance information extraction by discarding irrelevant blocks. The experimental results show that our method achieves state-of-the-art performance on the benchmark dataset. Further analysis demonstrates the effectiveness of our proposed modules and visually displays more compact semantic representations.

In the future, we might explore the following two research directions. On the one hand, we try to apply the framework to other computational argumentation tasks. On the other hand, we will explore the application of interactive argument identification in different fields, such as doctor consultation and student classroom discussion.

Limitations

There may be some possible limitations in this study. We observe a few arguments that express little information. Its subjects are primarily pronouns, in which case our ACE module may be limited. For example, an argument is “no offense, but that is incredibly stupid/selfish.”. Since the sentence expresses only a small amount of information, semantic similarity may not fully reflect the correlation between sentences, which affects the ACE module to some extent. In addition, although the performance is significantly improved after adding contrastive learning and the construction of the hard samples, it also increases the computational resources during the training process. In the future, we will design a more universal contextual enhancement module by introducing graph neural networks.

Acknowledgements

This research is supported by the National Natural Science Foundation of China (62077027), the program of China Scholarships Council (No.202007820024), the Ministry of Science and Technology of the People’s Republic of China (2018YFC2002500), and the Department of Science and Technology of Jilin Province, China (20200801002GH).

References

Tariq Alhindi and Debanjan Ghosh. 2021. " sharks are not the threat humans are": Argument component segmentation in school student essays. *arXiv preprint arXiv:2103.04518*.

Lauscher Anne, Ng Lily, Napoles Courtney, and Tetreault Joel. 2020. Rhetoric, logic, and dialectic: Advancing theory-based argument quality assessment in natural language processing. *COLING*, pages 4563–4574.

S. C. Christa Asterhan and B. Baruch Schwarz. 2007. The effects of monological and dialogical argumentation on concept learning in evolutionary theory. *JOURNAL OF EDUCATIONAL PSYCHOLOGY*, pages 626–639.

Teresa Botschen, Daniil Sorokin, and Iryna Gurevych. 2018. Frame- and entity-based knowledge for common-sense argumentative reasoning. *ArgMining@EMNLP*, pages 90–96.

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. *EMNLP 2020*.

Yung-Sung Chuang, Rumen Dangovski, Hongyin Luo, Yang Zhang, Shiyu Chang, Marin Soljačić, Shang-Wen Li, Wen-tau Yih, Yoon Kim, and James Glass. 2022. Diffcse: Difference-based contrastive learning for sentence embeddings. *arXiv preprint arXiv:2204.10298*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *north american chapter of the association for computational linguistics*.

Ming Ding, Chang Zhou, Hongxia Yang, and Jie Tang. 2020. Cogltx: Applying bert to long texts. *Advances in Neural Information Processing Systems*, 33:12792–12804.

Subhabrata Dutta, Jeevesh Juneja, Dipankar Das, and Tanmoy Chakraborty. 2022. Can unsupervised knowledge transfer from social discussions help argument mining? *arXiv preprint arXiv:2203.12881*.

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

Hongchao Fang, Sicheng Wang, Meng Zhou, Jiayuan Ding, and Pengtao Xie. 2020. Cert: Contrastive self-supervised learning for language understanding. *arXiv preprint arXiv:2005.12766*.

Andrea Galassi, Marco Lippi, and Paolo Torroni. 2018. Argumentative link prediction using residual networks and multi-objective learning. *ArgMining@EMNLP*, pages 1–10.

Andrea Galassi, Marco Lippi, and Paolo Torroni. 2021. Multi-task attentive residual networks for argument mining. *arXiv preprint arXiv:2102.12227*.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

Beliz Gunel, Jingfei Du, Alexis Conneau, and Ves Stoyanov. 2020. Supervised contrastive learning for pretrained language model fine-tuning. *arXiv preprint arXiv:2011.01403*.

Ivan Habernal, Henning Wachsmuth, Iryna Gurevych, and Benno Stein. 2018. The argument reasoning comprehension task: Identification and reconstruction of implicit warrants. *NAACL-HLT*, pages 1930–1940.

Lu Ji, Zhongyu Wei, Xiangkun Hu, Yang Liu, Qi Zhang, and Xuanjing Huang. 2018. Incorporating argument-level interactions for persuasion comments evaluation using co-attention model. *COLING*, pages 3703–3714.

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.

Alex Kendall, Yarin Gal, and Roberto Cipolla. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7482–7491.

Al Khalid Khatib, Yufang Hou, Henning Wachsmuth, Charles Jochim, Francesca Bonin, and Benno Stein. 2020. End-to-end argumentation knowledge graph construction. *national conference on artificial intelligence*.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33:18661–18673.

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

Minghan Li and Eric Gaussier. 2021. Keybld: Selecting key blocks with local pre-ranking for long document information retrieval. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2207–2211.

Minghan Li, Diana Nicoleta Popa, Johan Chagnon, Yagmur Gizem Cinar, and Eric Gaussier. 2021. The power of selecting key blocks with local pre-ranking for long document information retrieval. *arXiv preprint arXiv:2111.09852*.

Ilya Loshchilov and Frank Hutter. 2018. Fixing weight decay regularization in adam.

Ji Lu, Wei Zhongyu, Li Jing, Zhang Qi, and Huang Xuanjing. 2021. Discrete argument representation learning for interactive argument pair identification. *NAACL-HLT*, pages 5467–5478.

Anastasios Lytos, Thomas Lagkas, Panagiotis Sarigiannidis, and Kalina Bontcheva. 2019. The evolution of argumentation mining: From models to social media and emerging tools. *Information Processing & Management*, 56(6):102055.

Edward Ma. 2019. Nlp augmentation. https://github.com/makcedward/nlpaug.

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

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. *ACL*, pages 3259–3266.

Vlad Niculae, Joonsuk Park, and Claire Cardie. 2017. Argument mining with structured svms and rnns. *ACL*.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32:8026–8037.

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

Gabriella Skitalinskaya, Jonas Klaff, and Henning Wachsmuth. 2021. Learning from revisions: Quality assessment of claims in argumentation at scale. *EACL*, pages 1718–1729.

Christian Stab and Iryna Gurevych. 2014. Annotating argument components and relations in persuasive essays. *COLING*, pages 1501–1510.

Chenhao Tan, Vlad Niculae, Cristian Danescu-Niculescu-Mizil, and Lillian Lee. 2016. Winning arguments: Interaction dynamics and persuasion strategies in good-faith online discussions. *WWW*, pages 613–624.

Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(11).

Feng Wang and Huaping Liu. 2021. Understanding the behaviour of contrastive loss. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2495–2504.

Zhongyu Wei, Yang Liu, and Yi Li. 2016. Is this post persuasive? ranking argumentative comments in online forum. *ACL*.

Thomas Wolf, Julien Chaumond, Lysandre Debut, Victor Sanh, Clement Delangue, Anthony Moi, Pierric Cistac, Morgan Funtowicz, Joe Davison, Sam Shleifer, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.

Xing Wu, Chaochen Gao, Liangjun Zang, Jizhong Han, Zhongyuan Wang, and Songlin Hu. 2021. Esimcse: Enhanced sample building method for contrastive learning of unsupervised sentence embedding. *arXiv preprint arXiv:2109.04380*.

Jian Yuan, Zhongyu Wei, Yixu Gao, Wei Chen, Yun Song, Donghua Zhao, Jinglei Ma, Zhen Hu, Shaokun Zou, Donghai Li, et al. 2021a. Overview of smp-cail2020-argmine: The interactive argument-pair extraction in judgement document challenge. *Data Intelligence*, 3(2):287–307.

Jian Yuan, Zhongyu Wei, Donghua Zhao, Qi Zhang, and Changjian Jiang. 2021b. Leveraging argumentation knowledge graph for interactive argument pair identification. *ACL/IJCNLP*, pages 2310–2319.

A Block Segmentation Algorithm

Algorithm 1: Block Segmentation

```
Input: Context $c$, Punctuation costs $cost$,
       basic cost $co$, max block size $\alpha$
Initialize $f[0] ... f[\alpha - 1]$ as 0;
Initialize $from[0] ... from[\alpha - 1]$ as $-1$;
for $i$ $from$ $\alpha$ $to$ $len(c) - 1$ do
    $f[i] = +\infty$;
    for $j$ $form$ $i - \alpha$ $to$ $i - 1$ do
        if $word$ $is$ $punctuation$ then
            $v = cost[word] + f[j]$;
        else
            $v = co + f[j]$;
        if $v < f[i]$ then
            $f[i] = v, from[i] = j$
$t = len(c) - 1, blocks = []$;
while $t \geq 0$ do
    prepend $c[from[t] + 1 ... t]$ to blocks .
    $t = from[t]$
return $blocks$
```

B Hyperparameter Sensitivity Analysis

In this section, we investigate the impact of the two hyperparameters on our method. α (detailed in section 3.2.1) is the block size in context block segmentation module. It not only affects the result of similarity calculation between quotation/reply and each block but also determines the number of blocks input to the model because of the length limitation of BERT. τ (detailed in section 3.3.3) is an scalar temperature hyperparameter that controls the separation of classes. The following is the specific analysis of the two hyperparameters.

B.1 The Effect of α on Performance

To exploring the impact of α, we set the value of $\alpha \in \{16, 32, 42, 48, 64\}$. Accordingly, the number of input blocks $num \in \{16, 8, 6, 6, 4\}$ because the input length limitation of BERT. The results are shown in Table 8. Here we explain how to set the number of blocks. In the theory, the number and size of input blocks should satisfy the equation 5. However, in practice, we observe that the size of each block is often smaller than the block size we set because the length of each sentence is uncertain. For example, we set $\alpha = 64$. In practice, the length of most of the blocks is less than 64. Therefore, when setting the number of blocks, we should satisfy $\alpha \times num \approx 256$. num denotes number of blocks in the input. The experimental results are shown in the Table 8. Note Table 8 shows the results of the experiment before the introduction of contrastive learning. When $num = 6, \alpha = 42$, both metrics achieve the best results. We try to explain the phenomenon. When α is very small, the continuity of the sentences is limited, resulting in incoherent semantic information. When α is very large, the excessive block size will inevitably lead to irrelevant information in the sentence blocks, which affects the identification of the model. In addition, compared to $num = 6, \alpha = 42$ and $num = 4, \alpha = 64$, $num = 6, \alpha = 48$ has a significant performance degradation. We consider that the input truncation causes information loss because of $6 \times 48 = 288 \gg 256$. Therefore, the optimal combination is actually a trade-off, and in other experiments, we use the $num = 6, \alpha = 42$.

	P@1(%)	**MRR(%)**
$\tau = 0.03$	80.62	88.97
$\tau = 0.05$	**81.03**	**88.78**
$\tau = 0.1$	80.69	88.67
$\tau = 0.15$	79.95	88.38
$\tau = 0.2$	79.81	88.30
$\tau = 0.4$	79.68	88.24
$\tau = 0.6$	79.34	88.10
$\tau = 0.8$	79.09	87.88

Table 7: The results with different τ. The best results are highlighted in bold.

	P@1(%)	**MRR(%)**
$num = 16, \alpha = 16$	73.13	83.97
$num = 8, \alpha = 32$	74.41	84.82
$num = 6, \alpha = 42$	**75.11**	**85.67**
$num = 6, \alpha = 48$	74.14	84.57
$num = 4, \alpha = 64$	75.08	85.08

Table 8: The results with different num, α. The best results are highlighted in bold.

B.2 The Effect of τ on Performance

As mentioned by (Wang and Liu, 2021; Gunel et al., 2020), contrastive loss is a hardness-aware loss. τ controls the strength of penalties on hard negative samples. Small τ tends to generate more uniform distribution and be less tolerant to similar samples. In this section, we explore the impact of τ on this task. We set the value of

$\tau \in \{0.03, 0.05, 0.1, 0.15, 0.2, 0.4, 0.6, 0.8\}$. The results are shown in Table 7. From the experimental results, $\tau = 0.05$ is the optimal hyperparameter. Besides, with the increase of T, the experimental results become worse and worse. In other experiments, we use $\tau = 0.05$.

C Error Analysis

We observe two main problems with our methods for some instances of wrong predictions:

- As mentioned in above, a few arguments that express little information and whose subjects are primarily pronouns, in which case our ACE module may be limited. For example, an argument is "no offense, but that is incredibly stupid/selfish.". Since the sentence expresses only a small amount of information, semantic similarity may not fully reflect the correlation between sentences, which affects the ACE module to some extent, which may affect the ACE module to some extent. In this case, it might be better to use the adjacent context block directly.
- In addition, some contexts are relatively short, even less than 200 words. At this time, the ACE module uses all the contexts as the input of the model and may add some information that is not relevant to the argument, which is one of the reasons for the wrong prediction of the model.