Towards Better Document-level Relation Extraction via Iterative Inference

Liang Zhang[1,2*]   Jinsong Su[1,2*]   Yidong Chen[1,2†]   Zhongjian Miao[1,2]   Zijun Min[1,2]
Qingguo Hu[1,2]   Xiaodong Shi[1,2]

[1]School of Informatics, Xiamen University, China

[2]Key Laboratory of Digital Protection and Intelligent Processing of Intangible Cultural Heritage of Fujian and Taiwan (Xiamen University), Ministry of Culture and Tourism, China

lzhang@stu.xmu.edu.cn   {jssu,ydchen}@xmu.edu.cn

Abstract

Document-level relation extraction (RE) aims to extract the relations between entities from the input document that usually containing many difficultly-predicted entity pairs whose relations can only be predicted through relational inference. Existing methods usually directly predict the relations of all entity pairs of input document in a one-pass manner, ignoring the fact that predictions of some entity pairs heavily depend on the predicted results of other pairs. To deal with this issue, in this paper, we propose a novel document-level RE model with iterative inference. Our model is mainly composed of two modules: 1) *a base module* expected to provide preliminary relation predictions on entity pairs; 2) *an inference module* introduced to refine these preliminary predictions by iteratively dealing with difficultly-predicted entity pairs depending on other pairs in an easy-to-hard manner. Unlike previous methods which only consider feature information of entity pairs, our inference module is equipped with two *Extended Cross Attention* units, allowing it to exploit both feature information and previous predictions of entity pairs during relational inference. Furthermore, we adopt a two-stage strategy to train our model. At the first stage, we only train our base module. During the second stage, we train the whole model, where contrastive learning is introduced to enhance the training of inference module. Experimental results on three commonly-used datasets show that our model consistently outperforms other competitive baselines. Our source code is available at https://github.com/DeepLearnXMU/DocRE-II.

1 Introduction

Relation extraction (RE) aims to identify the relation between two entities from raw texts. Due to its wide applications in many subsequent natural
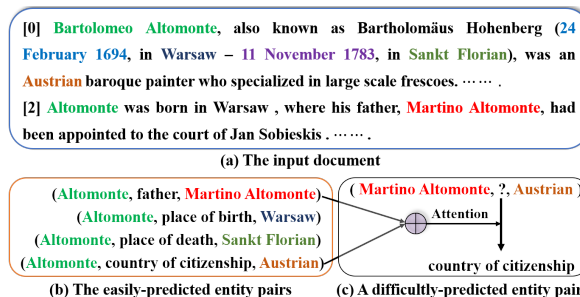
*Equal Contribution.
†Corresponding Author.

Figure 1: An example comes from the DocRED dataset. (a) is an input document, where different colors represent different entities. (b) lists some easily-predicted entity pairs whose predictions do not require reference to the predictions of other pairs. (c) shows a difficultly-predicted entity pair, of which prediction depends on the predicted results of its overlapping pairs. The arrows between (b) and (c) indicate the dependencies among entity pairs, which can be exploited to benefit the relation prediction of the difficultly-predicted one.

language processing (NLP) tasks, such as large-scale knowledge graph construction (Zeng et al., 2020) and question answering (Yu et al., 2017), RE has attracted increasing attention and become a fundamental NLP task. Most of the existing works focus on sentence-level RE, where both considered entities come from a single sentence (Zhang et al., 2018; Baldini Soares et al., 2019). However, large amounts of relations, such as relational facts from Wikipedia articles, are expressed by multiple sentences in real-world applications (Verga et al., 2018; Zhou et al., 2021). As calculated by Yao et al. (2019), in the commonly-used DocRED dataset, the identifications of more than 40.7% relational facts involve multiple sentences. Therefore, a natural extension is document-level RE, which is required to exploit the input document to infer all relations between entities.

However, compared with sentence-level RE, document-level RE is a more challenging task. This is because each document often contains a large number of entity pairs whose relations need to be predicted. More importantly, the relation prediction

difficulties of these pairs are usually significantly different. The relations of some entity pairs can be directly predicted while there also exist many entity pairs whose relations can only be inferred by referring to other pairs. As calculated by Yao et al. (2019), about 61.6% entity pairs fall into the latter category in the DocRED dataset.

Figure 1 shows an example from the DocRED dataset, which contains totally five entity pairs with relations. Among these entity pairs, the relations of some pairs can be easily identified even without knowing relations of other pairs, such as (*Altomonte, Martino Altomonte*) and (*Altomonte, Austrian*) (See Figure 1b). By contrast, it is difficult to correctly identify the relation of (*Martino Altomonte, Austrian*) because the document does not contain sufficient evidence information for it. Furthermore, if the relations of (*Altomonte, Martino Altomonte*) and (*Altomonte, Austrian*) are predicted first, it will become easier to predict the relations of (*Martino Altomonte, Austrian*) with the help of these previously-predicted results (See Figure 1c).

To deal with the above issues, many researchers introduce graph neural networks (GNNs) (Kipf and Welling, 2017; Guo et al., 2019) to exploit the dependencies among entities or mentions for document-level RE (Christopoulou et al., 2019; Zeng et al., 2020; Nan et al., 2020; Wang et al., 2020a). However, these methods only focus on the dependencies at the entity- or mention-level while neglecting the dependencies among entity pairs, which has an important impact on the relation identification of difficultly-predicted entity pairs depending on other pairs. Further, Zhang et al. (2021) model document-slevel RE as a semantic segmentation problem. In this way, the dependencies among entity pairs can be captured via CNN, which, however, ignores the fact that relation prediction difficulties of entity pairs are different. Instead, Zeng et al. (2021) treat intra- and inter-sentence entity pairs as easily-predicted and difficultly-predicted ones, and use different encoders to learn their representations, respectively. Although this method is simple, it cannot accurately distinguish prediction difficulties of entity pairs.

In this paper, we propose a document-level RE model with iterative inference. Overall, as shown in Figure 2, our model mainly consists of two modules: 1) *an base module* used to firstly predict the relations of entity pairs roughly; 2) *a inference mod-*

ule exploiting the prediction results of the previous iteration to refine the predictions in an iterative manner. Particularly, we equip the inference module with an attention mechanism, which enables the module to accurately exploit the dependencies among overlapping entity pairs. By doing so, we expect that based on the prediction results of the previous iteration, inference module can deal with difficultly-predicted entity pairs depending on other pairs in an easy-to-hard manner.

Furthermore, we adopt a two-stage strategy to train our model. First, we only train our base module. Second, we train the entire model, where the base module is set with a relatively small learning rate to keep its parameters and performance stable. Particularly, at the second stage, we introduce contrastive learning to encourage inference module to exploit the previously-predicted results better.

To investigate the effectiveness of our model, we conduct several groups of experiments on commonly-used datasets. Experimental results and in-depth analyses strongly demonstrate the superiority of our model.

2 Our Proposed Model

In this section, we elaborate on our model. As shown in Figure 2, our model is composed of two modules: base module (Section 2.1) and inference module (Section 2.2). Finally, we give a detailed description of the model training (Section 2.3).

2.1 Base Module

We chose a competitive baseline model, ATLOP (Zhou et al., 2021), to construct our base module. It is used to make preliminary predictions on relations of entity pairs providing basic information for inference module. Here, we give a brief introduction to ATLOP. Please refer to (Zhou et al., 2021) for more details.

Let $D = [w_1, w_2, ..., w_L]$ denotes the input document that contains a set of entities $\{e_i\}_{i=1}^N$. Note that an entity e_i may appear multiple times in the document by mentions $\{m_j^i\}_{j=1}^{N_{e_i}}$. To obtain better entity representations, we first insert a special symbol "$*$" at the start and end of the mention to mark its span. Then, we feed the document into a pre-trained language model, obtaining its contextual embeddings: $H = [h_1, h_2, ..., h_L]$. Particularly, we take the context embedding of each mention's start symbol "$*$" as its feature vector. Finally, via *logsumexp* pooling (Jia et al., 2019), we
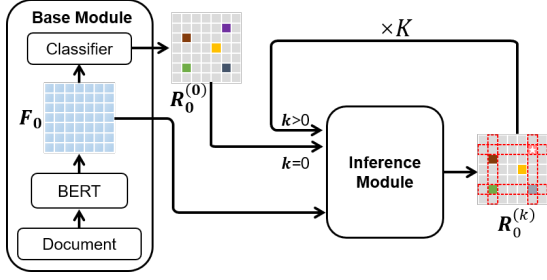
Figure 2: The overall architecture of our model. We first use base module to make preliminary predictions. Then, inference module gradually revises the preliminary predictions through K-iteration inference. F_0 and $R_0^{(*)}$ refer to the feature matrix and relation matrix of entity pairs, respectively. Note that the initial input of inference module is $(F_0, R_0^{(0)})$ (k=0), while the input of inference module becomes $(F_0, R_0^{(k)})$ in the subsequent iterations (k>0).

aggregate all mention-level context embeddings of entity e_i to obtain its final representation vector, *i.e.*, $h(e_i) = \log \sum_{j=1}^{N_{e_i}} \exp(h(m_j^i))$.

Unlike ATLOP directly using a bilinear function to predict the relations of entity pairs, we firstly compute a feature vector for each entity pair, and then stack a single linear layer to predict their relations. Specifically, for an entity pair (e_s, e_o), we calculate a local context vector $c_{s,o}$ according to the attention matrix A of the last BERT layer:

$$c_{s,o} = H^\mathsf{T} \frac{A_s \circ A_o}{\mathbf{1}^\mathsf{T}(A_s \circ A_o)}, \qquad (1)$$

where A_s and A_o denote the attention weights of entity e_s and e_o to all tokens in document, respectively, and $\circ$ refers to element-wise multiplication. By doing so, $c_{s,o}$ can effectively capture the local contextual information related to both entities, which can be further used to enhance the representation of entity pair (e_s, e_o). Afterwards, we compute the initial feature vector $F_{s,o}$ of (e_s, e_o) as

$$F_{s,o} = \mathrm{FNN}\big(\big[\tanh(W_s[h(e_s), c_{s,o}]), \qquad (2)$$
$$\tanh(W_o[h(e_o), c_{s,o}]) \big), \qquad (3)$$

where $\mathrm{FNN}(\cdot)$ refers to a feed-forward neural network, W_o and W_s are learnable weight matrices. Finally, we obtain the probability distribution $p_{s,o}$ of relations assigned to (e_s, e_o) through a simple linear layer:

$$p_{s,o} = \sigma(W_r F_{s,o} + b_r), \qquad (4)$$

where W_r and b_r are model parameters.

2.2 Inference Module

As mentioned previously, we introduce inference module to iteratively refine the predictions of base
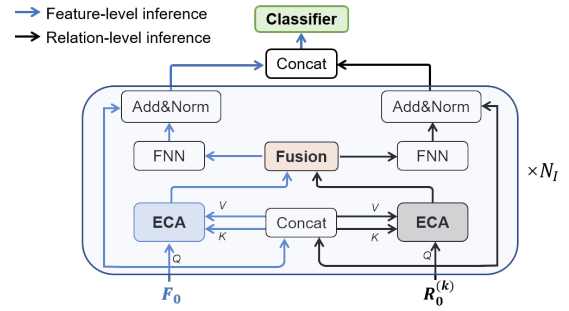

Figure 3: Illustration of Inference Module. It contains N_I inference layers and a classifier. With two ECA units, inference layers perform feature- and relation-level inference on the feature and relation matrices.

module, until the maximal iteration number K is reached. As illustrated in Figure 3, this module consists of N_I inference layers and a classifier. Inference layers are used to perform inference, where the dependencies among overlapping entity pairs are leveraged to learn better entity pair representations. Then, with the output of the top inference layer, the classifier produces better predictions.

To facilitate the computation of inference module, we combine the feature vectors of all entity pairs into a *feature matrix* $F_0 = [F_{s,o}]_{N \times N}$, where each row $F_{0[s,*]}$ corresponds to a subject entity e_s and each column $F_{0[*,o]}$ corresponds to an object entity e_o. Similarly, we employ the embedding operation $\mathrm{emb}(\cdot)$ to construct a *relation matrix* $R_0^{(0)} = [\mathrm{emb}(\arg\max(p_{s,o}))]_{N \times N}$ from the prediction results of base module, where $p_{s,o}$ is calculated in Equation (4), the subscript $*$ and superscript $(*)$ of $R_0^{(0)}$ denote the inference layer index and iterative index, respectively.

Inference Layer At the k-th iteration, with F_0 and $R_0^{(k)}$ as inputs, inference layers are committed to learning more expressive feature matrix F_{N_I} and relation matrix $R_{N_I}^{(k)}$. Back to Figure 3, each inference layer contains three core components: 1) two *Extended Cross Attention* (ECA) units performing feature- and relation-level inference, respectively (See the blue and black lines in Figure 3), and 2) a *Fusion sub-layer* combining the outputs of ECA units. Since inference layers perform the feature- and relation-level inference in the same way, we take the feature-level inference as an example to illustrate its details.

ECA is a variant of the conventional multi-head self-attention. The basic intuition behind ECA is that for each entity pair (e_s, e_o), its overlapping entity pairs can provide important information for inferring its relations. To model this intuition, we

extend the multi-head self-attention to ECA that only focuses on overlapping entity pairs. Note that these overlapping entity pairs are only located in the s-th row, s-th column, o-th row and o-th column of the feature matrix. To effectively exploit their information, we equip ECA with four attention heads to capture their effects, respectively. For example, at the l-th layer, we calculate the first attention head of ECA as follows:

$$\text{head}_1 = \text{Attention}(F_{l[s,o]}, M_{l[s,*]}, M_{l[s,*]}), \quad (5)$$

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^\mathsf{T}}{\sqrt{d_k}})V, \quad (6)$$

where $M_l = [F_l, R_l^{(k)}]$ is the concatenated matrix of F_l and $R_l^{(k)}$, $M_{l[s,*]}$ denotes the s-th row of M_l, d_k is the dimension of key (K) and query (Q). Then, we merge all attention heads to obtain the output of ECA, which is a temporary feature matrix $\widetilde{F}_l$:

$$\widetilde{F}_l = \text{Concat}(\text{head}_1, ..., \text{head}_4)W_O, \quad (7)$$

where W_O is a model parameter matrix. Likewise, we use the other ECA to obtain a temporary relation matrix $\widetilde{R}_l^{(k)}$. Subsequently, fusion sub-layer combines $\widetilde{F}_l$ and $\widetilde{R}_l^{(k)}$ through a gating mechanism:

$$s_l = \text{Sigmoid}([\widetilde{F}_l, \widetilde{R}_l^{(k)}]W_g + b_g), \quad (8)$$

$$\widetilde{M}_l = s_l \circ \widetilde{F}_l + (1 - s_l) \circ \widetilde{R}_l^{(k)}, \quad (9)$$

where W_g and b_g are trainable parameters.

Finally, we obtain the output of the l-th inference layer as follows:

$$F_{l+1} = \text{LayerNorm}(F_l + \text{FNN}(\widetilde{M}_l)), \quad (10)$$

$$R_{l+1}^{(k)} = \text{LayerNorm}(R_l^{(k)} + \text{FNN}(\widetilde{M}_l)), \quad (11)$$

where $\text{LayerNorm}(\cdot)$ is layer normalization (Ba et al., 2016).

After repeating this inference process N_I times, we get the updated feature matrix F_{N_I} and relation matrix $R_{N_I}^{(k)}$.

Classifier On the basis of $R_{N_I}^{(k)}$ and F_{N_I}, we stack a single-layer classifier to provide more refined predictions on entity pairs:

$$P^{(k+1)} = \sigma(W_c[F_{N_I}, R_{N_I}^{(k)}] + b_c). \quad (12)$$

Finally, we obtain the relation matrix of the $(k+1)$-th iteration: $R_0^{(k+1)} = \text{emb}(\arg\max(P^{(k+1)}))$.

We perform the above-mentioned inference for K times, obtaining the final predictions $P^{(K)}$.

2.3 Model Training

We adopt a two-stage strategy to train our model. At the first stage, we use an *adaptive threshold loss*
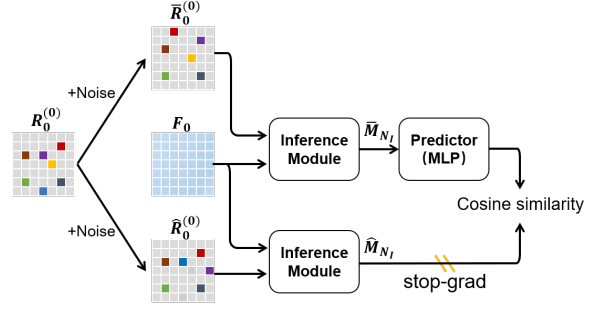

Figure 4: Illustration of our contrastive learning. The stop-gradient operation *stop-grad* can effectively prevent the representation space of the model from collapsing. Note the feature matrix and relation matrix fed into the two inference modules sharing parameters are in reverse order, which forces the inference module to treat both types of matrices equally.

$\mathcal{L}_\mathcal{R}$ to only train base module. At the second stage, we train the entire model, where we set a relatively small learning rate for base module to maintain its parameters and performance stable. Particularly, in addition to $\mathcal{L}_\mathcal{R}$, we introduce a *contrastive loss* $\mathcal{L}_\mathcal{C}$ to enhance the training of inference module. To improve the training efficiency at this stage, following Ghazvininejad et al. (2019), we train inference module to directly correct the predictions of base module in a one-pass manner, as opposed to the multiple iterations used during testing. Next, we describe our losses in detail.

Adaptive Threshold Loss $\mathcal{L}_\mathcal{R}$ This loss is proposed by Zhou et al. (2021), aiming to alleviate the imbalanced relation distribution problem in document-level RE. In this work, we introduce this loss into our model training, which is defined as

$$\mathcal{L}_\mathcal{R} = -\sum_{r \in \mathcal{P}_T} \log\left(\frac{\exp(\text{logit}_r)}{\sum_{r' \in \{\mathcal{P}_T, \text{TH}\}} \exp(\text{logit}_{r'})}\right)$$
$$- \log\left(\frac{\exp(\text{logit}_\text{TH})}{\sum_{r' \in \{\mathcal{N}_T, \text{TH}\}} \exp(\text{logit}_{r'})}\right), \quad (13)$$

where TH is a threshold relation used to distinguish between positive and negative relations. This loss will push the logits of all positive relations $\mathcal{P}_T$ to be higher than that of TH, and pull the logits of all negative relations $\mathcal{N}_T$ to be lower than that of TH.

Contrastive Loss $\mathcal{L}_\mathcal{C}$ To prevent inference module from just simply replicating the predictions of base module, we inject noises into the relation matrix $R_0^{(0)}$ by randomly substituting r percent of predicted relations with incorrect ones. However, such a noises injection renders the relation matrix $R_0^{(0)}$ less stable than the feature matrix F_0. As a result, inference module prefers predicting relations

using only F_0. To address this defect, we introduce a variant method of contrastive learning, SimSiam (Chen and He, 2021), to enhance the training of our inference module. Unlike the conventional contrastive learning, such as (Chen et al., 2020a; He et al., 2020; Khosla et al., 2020), it only focuses on pulling together the representations of examples in *positive pairs*. In this way, it can avoid the drawback of pushing *negative pairs* far apart in conventional contrastive learning, which may limit the potential of inference module to capture the dependencies among entity pairs.

Concretely, as shown in Figure 4, we generate two different relation matrices, $\overline{R}_0^{(0)}$ and $\widehat{R}_0^{(0)}$, by adding various noises to relation matrix $R_0^{(0)}$. Then, we obtain a positive pair consisting of two examples $(F_0, \overline{R}_0^{(0)})$ and $(\widehat{R}_0^{(0)}, F_0)$, which are fed into inference module to produce the corresponding outputs $\overline{M}_{N_I}$ and $\widehat{M}_{N_I}$, respectively. Please note that the input order of the feature and relation matrices is inverted in these two examples. Finally, we define the contrastive loss as follows:

$$\mathcal{L}_{\mathcal{C}} = 2 - \Big(\mathrm{Cosine}\big(\mathrm{MLP}(\widehat{M}_{N_I}), \mathrm{SG}(\overline{M}_{N_I})\big) \\ + \mathrm{Cosine}\big(\mathrm{MLP}(\overline{M}_{N_I}), \mathrm{SG}(\widehat{M}_{N_I})\big) \Big), \quad (14)$$

where $\overline{M}_{N_I} = [F_{N_I}, \overline{R}_{N_I}^{(0)}]$ and $\widehat{M}_{N_I} = [F_{N_I}, \widehat{R}_{N_I}^{(0)}]$ are the outputs of the top inference layer, $\mathrm{Cosine}(\cdot)$ indicates a cosine similarity function, $\mathrm{SG}(\cdot)$ refers to a stop-gradient operation that prevents the model training from collapsing (Chen and He, 2021), and $\mathrm{MLP}(\cdot)$ stands for a multi-layer perceptron function, our predictor, which helps the model learn better representations (Chen and He, 2021; Chen et al., 2020a). In this way, inference module will be encouraged to fully utilize the relation matrix.

3 Experiments

3.1 Datasets

DocRED (Yao et al., 2019) This dataset is a large-scale crowdsourced dataset for document-level RE, which is constructed from Wikipedia and Wikidata. It contains 3,053 documents for training, 1,000 for development, and 1,000 for testing. Each document contains 26 entities on average. In total, this dataset involves 97 target relations.

CDR (Li et al., 2016) It is a biomedical dataset that is constructed from the PubMed abstracts. CDR has only one target relation: *Chemical-Induced-Disease* between chemical and disease entities. It includes about 1,500 human-annotated documents, which are equally split into training, development and test sets.

GDA (Wu et al., 2019) It is also a biomedical dataset, which is constructed from MEDLINE abstracts via distant supervision. Following Christopoulou et al. (2019), we split its training set into two parts: training set (23,353 documents) and development set (5,839 documents), and directly evaluate the model on its test set (1,000 documents).

3.2 Settings

We develop the proposed model based on PyTorch. We used BERT-base (Devlin et al., 2019) or RoBERTa-large (Liu et al., 2019) as the encoder on DocRED and SciBERT (Beltagy et al., 2019) on CDR and GDA. Inspired by Ghazvininejad et al. (2019), we sample the noise rate r of the relation matrix from a uniform distribution $U(0, 0.4)$ during training. We apply AdamW (Loshchilov and Hutter, 2019) to optimize our model, with a linear warmup (Goyal et al., 2017) during the first 6% steps followed by a linear decay to 0. All hyperparameters are tuned on the development set, and some of them are listed in Appendix A.

3.3 Baselines

We compare our model with the following two types of baselines:

Graph-based Models These models first construct a document graph from the input document and then perform inference on the graph through GNNs. We include EoG (Christopoulou et al., 2019), DHG (Zhang et al., 2020), GEDA (Li et al., 2020), LSR (Nan et al., 2020), GLRE (Wang et al., 2020a), GAIN (Zeng et al., 2020), HeterGSAN (Xu et al., 2021), and SSAN (Xu et al., 2021) for comparison.

Transformer-based Models These models directly use the pre-trained language models for document-level RE without graph structures. We compare BERT-base (Wang et al., 2019a), BERT-TS (Wang et al., 2019a), HIN-BERT (Tang et al., 2020), Coref-BERT (Ye et al., 2020), and ATLOP-BERT (Zhou et al., 2021) with our model.

Besides, we consider some recently-proposed methods that exploit the dependencies among entity pairs to improve the performance of document-level RE, including DocuNet (Zhang et al., 2021), SIRE (Zeng et al., 2021), and KD (Tan et al., 2022).

Model	Dev				Test	
	$\text{Ign}F_1$	F_1	Intra-$F1$	Inter-$F1$	$\text{Ign}F_1$	F_1
GEDA-BERT (Li et al., 2020)	54.52	56.16	–	–	53.71	55.74
LSR-BERT (Nan et al., 2020)	52.43	59.00	65.26	52.05	56.97	59.05
GLRE-BERT (Wang et al., 2020a)	–	–	–	–	55.40	57.40
GAIN-BERT (Zeng et al., 2020)	59.14	61.22	67.10	53.90	59.00	61.24
HeterGSAN-BERT (Xu et al., 2021)	58.13	60.18	–	–	57.12	59.45
SSAN-BERT (Xu et al., 2021)	56.68	58.95	–	–	56.06	58.41
BERT (Wang et al., 2019a)	–	54.16	61.61	47.15	–	53.20
BERT-Two-Step (Tang et al., 2020)	–	54.42	61.80	47.28	–	53.92
HIN-BERT (Tang et al., 2020)	54.29	56.31	–	–	53.70	55.60
CorefBERT (Ye et al., 2020)	55.32	57.51	–	–	54.54	56.96
ATLOP-BERT (Zhou et al., 2021)	59.22	61.09	–	–	59.31	61.30
DocuNet-BERT (Zhang et al., 2021)	59.86	61.83	–	–	59.93	61.86
SIRE-BERT (Zeng et al., 2021)	59.82	61.60	68.07	54.01	60.18	62.05
KD-BERT (Tan et al., 2022)	60.08	62.03	–	–	60.04	62.08
Ours-BERT	**60.75**±0.12	**62.74**±0.15	**69.14**±0.10	**55.54**±0.19	**60.68**	**62.65**

Table 1: The model performance on the development and test sets of DocRED. We run experiments 5 times with different random seeds and report the mean and standard deviation on the development set. We save the best checkpoint on the development set and then report the official test scores on the CodaLab scoreboard. The results of RoBERTa-large-based model are reported in Appendix B.

K/N_I	1	2	3	4
1	61.53	61.85	61.80	61.66
2	61.96	62.29	62.07	62.01
3	62.41	62.74	62.66	62.47
4	62.25	62.65	62.53	62.33

Table 2: The performance (F_1 points) of our model with different values of K and N_I on the development set of DocRED.

3.4 Effect of Iteration Number K and Layer Number N_I

The iteration number K of inference module and the layer number N_I of inference layers are two important hyper-parameters of our model, which directly affect the performance of inference module. Thus, we conduct an experiment with different values of K and N_I on the development set of DocRED. From Figure 5, we observe that our model achieves the best performance when K and N_I are set to 3 and 2, respectively. Hence, we use K=3 and N_I=2 in all subsequent experiments.

3.5 Main Results

Results on DocRED Following Zeng et al. (2020), we use F_1 and $\text{Ign}F_1$ as the evaluation metrics. $\text{Ign}F_1$ denotes the $F1$ points excluding the relational facts that are shared by the training and developmen/test sets. As shown in Table 1, our model consistently outperforms all baselines. Besides, we draw the following interesting conclusions:

First, our model performs better than ATLOP-BERT, our base model, by **1.35** F_1 and **1.37** $\text{Ign}F_1$ points on the test set, which demonstrates the effectiveness of our inference module.

Second, our model also obtains improvements of **1.41** F_1 and **1.86** $\text{Ign}F_1$ points on the test set, compared with the graph-based SOTA model, GAIN-BERT, which exploits the entity- and mention-level dependencies for document-level RE. These results demonstrate that the dependencies among entity pairs are more important for document-level RE than ones among entities or mentions.

Third, our model surpasses SIRE-BERT and KD-BERT, both of which use an attention mechanism to capture the dependencies among entity pairs. This verifies that our model can more effectively capture the dependencies among entity pairs.

Finally, we follow Zeng et al. (2020, 2021) to report Intra-F_1 and Inter-F_1 points in Table 1. Please note that these two metrics only consider intra- and inter-sentence relations, respectively. Compared with Intra-F_1, Inter-F_1 can better reflect the inference ability of the model. In terms of Inter-F_1, our model surpasses SIRE-BERT by **1.53** points.

Results on the Biomedical Datasets We also conduct experiments on the biomedical datasets, of which results are shown in Table 3. Our model still consistently outperforms all previous baselines. On CDR and GDA, our model obtains F_1 points of **73.2** and **85.9**, with absolute improvements of **3.8** and **2.0** over our base model (ATLOP-SciBERT), respectively. Thus, we confirm that our model is

Model	CDR	GDA
BRAN (Verga et al., 2018)	62.1	–
EoG (Christopoulou et al., 2019)	63.6	81.5
LSR (Nan et al., 2020)	64.8	82.2
DHG (Zhang et al., 2020)	65.9	83.1
SciBERT (Beltagy et al., 2019)	65.1	82.5
ATLOP-SciBERT (Zhou et al., 2021)	69.4	83.9
SIRE-BioBERT (Zeng et al., 2021)	70.8	84.7
Ours-SciBERT	**73.2**	**85.9**

Table 3: The F_1 points on the CDR and GDA test sets.

also genralized to biomedicine.

3.6 Ablation Study

To investigate the effectiveness of different components on our model, we further compare our model with the following variants in Table 4.

(1) *w/o fusion sub-layer*. In this variant, we remove the fusion sub-layer from inference module, which leads to a drop of **0.37** F_1 points. It suggests that combining the feature- and relation-level inference information of entity pairs is indeed useful for improving the performance of the model.

(2) *w/o ECA*. In this variant, we replace each ECA unit with a standard multi-head self-attention, where all other entity pairs can be considered. This change causes a significant performance decline. The underlying reason is that focusing on all other entity pairs introduces many noises to our model.

(3) *w/o contrastive loss $\mathcal{L}_\mathcal{C}$*. When we discard the contrastive loss during the second training stage, the performance of our model degrades by **0.58** F_1 points, which confirms that our contrastive loss effectively enhances inference module. Inspired by Wu et al. (2021b), we examine the average weights of the feature matrix and relation matrix in the classifier of the inference module, which can intuitively reflect the importance of the two matrices in relation prediction (See W_c in Equation (12)). From Figure 5(a), we observe that inference module prefers to predict relations using the more stable feature matrix. However, when introducing the contrastive loss $\mathcal{L}_\mathcal{C}$, inference module can simultaneously exploit both feature matrix and relation matrix in relation prediction (See Figure 5(b)). Specifically, in this variant, the average weights of the feature matrix and the relation matrix are 0.045 and 0.007, while they are 0.026 and 0.023 in our model. These results suggest that our contrastive loss can encourage inference module to better exploit the relation matrix.
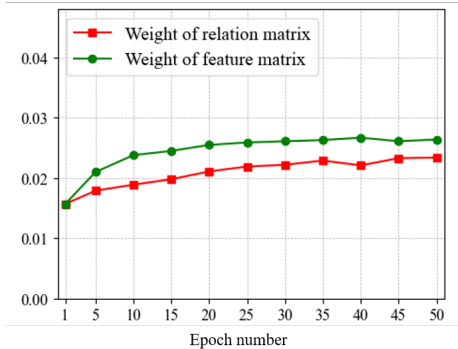
(4) *w/ negative pairs*. In this variant, we replace

Model	IgnF_1	F_1
ATLOP-BERT (our base)	59.22	61.09
Ours-BERT	**60.75**	**62.74**
w/o fusion sub-layer	60.43	62.37
w/o ECA	59.29	61.22
w/o contrastive loss $\mathcal{L}_\mathcal{C}$	60.26	62.16
w/ negative pairs	60.25	62.06
w/o pre-training	59.51	61.45
w/ freeze base module	60.35	62.31

Table 4: Ablation study of our model on the development set of DocRED.
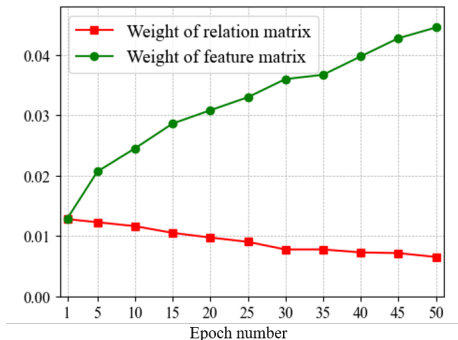
(a) *w/o* **contrastive loss $\mathcal{L}_\mathcal{C}$**

(b) **Our model**

Figure 5: Illustration of the average weights of feature matrix and relation matrix in the classifier of the inference module.

our contrastive learning with SimCLR (Chen et al., 2020a) and take two different entity pairs from the same entity pair matrix to produce negative pairs. In addition to pulling together representations of samples in positive pairs like our contrastive learning, this variant also pushes the representations of samples in negative pairs far apart. Apparently, the performance drop reported in line 7 indicates that pushing entity pairs far apart limits the potential of inference module to capture the dependencies among entity pairs.

(5) *w/o pre-training*. Different from our two-stage training strategy, this variant directly trains our entire model without the pre-training of base module. From the line 8 of Table 4, we can observe a drop of **1.29** F_1 points, confirming that the pre-

Model	Dev		Test	
	$\mathrm{Ign}F_1$	F_1	$\mathrm{Ign}F_1$	F_1
Coref-RoBERTa-large (Ye et al., 2020)	57.35	59.43	57.90	60.25
GAIN-RoBERTa-large (Zeng et al., 2020)	60.87	63.09	60.31	62.76
SSAN-RoBERTa-large (Xu et al., 2021)	59.40	61.42	60.25	62.08
ATLOP-RoBERTa-large (Zhou et al., 2021)	61.32	63.18	61.39	62.40
DocuNet-RoBERTa-large (Zhang et al., 2021)	62.23	64.12	62.39	64.55
KD-RoBERTa-large (Tan et al., 2022)	62.16	64.19	62.57	64.28
Ours-RoBERTa-large	**62.66**±0.11	**64.58**±0.13	**62.92**	**64.88***

Table 5: The performance of the RoBERTa-large-based model on the development and test sets of DocRED. * denotes significant at $\rho<0.01$ with 1,000 bootstrap tests. We run experiments 5 times with different random seeds and report the mean and standard deviation on the development set. We save the best checkpoint on the development set and then report the official test scores on the CodaLab scoreboard.

Model	Infer-F_1	Pre.	Recall
GAIN-GloVe	40.82	32.76	54.14
SIRE-GloVe	42.72	34.83	55.22
BERT-RE	39.62	34.12	47.23
GAIN-BERT	46.89	38.71	59.45
Ours-BERT	**48.75**	**45.02**	**53.15**
w/o ECA	46.91	41.05	54.73
w/o Inference module	46.76	38.74	58.96

Table 6: Infer-F_1 points on the dev set of DocRED.

trained base module can provide inference module with better basic information.

(6) *w/ freeze base module.* To confirm that the performance improvement of our model mainly benefits from inference module, we freeze base module at the second stage of training. This variant still achieves an F_1 score of **62.31**, which improves the performance of our base model (ATLOP-BERT) by **1.22** F1 points. This also implies that our inference module can be used with other types of base modules in a plug-and-play manner.

3.7 Analysis of Inference Performance

To further evaluate the inference ability of our model, we follow Zeng et al. (2020, 2021) to report Infer-F_1 points in Table 6, which only considers the relations engaged in the relational inference process. For example, if the relational triples (e_h, r_1, e_o), (e_o, r_2, e_t) and (e_h, r_3, e_t) co-occur in the same document, we take them into account in the calculation of Infer-F_1 points.

In terms of Infer-F_1, our model yields an improvement of **1.86** points over GAIN. Moreover, the performance of our model sharply drops by **1.84** Infer-F_1 points when we replace ECA units with standard multi-head self-attentions. Meanwhile, without inference module, our model also suffers from performance degradation of **1.99** Infer-

F_1 points. All these results also strongly demonstrate that our inference module can effectively improve the inference ability of the model.

Finally, we introduce a case study in Appendix B to visually show the effectiveness of our model.

3.8 RoBERTa-large-based model

Following some recent studies(Zhou et al., 2021; Zhang et al., 2021; Tan et al., 2022), we also report the performance of the RoBERTa-large based models on the DocRED dataset. From Table 5, we can observe that our model consistently outperforms all baselines. Specifically, our model significantly surpasses our base module(ATLOP-RoBERTa-large) by 1.48 $F1$ points (statistical significance $\rho<0.01$), and also exceeds KD-RoBERTa-large by 0.6 $F1$ points on the test sets of DocRED.

4 Related Work

Early studies on RE mainly focus on sentence-level RE, which predicts the relation between two entities within a single sentence. In this aspect, many approaches (Zeng et al., 2015; Feng et al., 2018; Wang et al., 2020b; Ye et al., 2020; Yu et al., 2020; Wu et al., 2021a; Huang et al., 2021; Li et al., 2021) have been proven to be effective in this task. However, because many relational facts in real applications are expressed by multiple sentences (Yao et al., 2019), researchers gradually shift their attention to document-level RE.

To this end, researchers have proposed two kinds of methods: Transformer-based and GNN-based methods. Due to the fact that GNNs can model the dependencies among entities or mentions and have strong inference ability, many researchers explore GNNs for better document-level RE (Christopoulou et al., 2019; Li et al., 2020; Zhang et al., 2020; Zhou et al., 2020; Wang et al.,

2020a; Nan et al., 2020; Zeng et al., 2020; Xu et al., 2021). Usually, they first construct a document graph, which uses mentions or entities as nodes and leverages heuristic rules and semantic dependencies to build edges. Then, they perform inference with GNNs on the graph. For example, Nan et al. (2020) treat the document graph as a latent variable which can be dynamically induced via structure attention. During this process, the induced graph structure can be exploited for better inference in document-level RE. Zeng et al. (2020) propose a graph aggregation-and-inference network involving a heterogeneous mention-level graph and an entity-level graph. These two graphs are utilized to model dependencies among mentions and entities, respectively. Meanwhile, due to the advantage of Transformer (Vaswani et al., 2017) on implicitly modeling long-distance dependencies, some studies (Wang et al., 2019b; Tang et al., 2020; Zhou et al., 2021) directly apply pre-trained language models to document-level RE. Zhou et al. (2021) propose the ATLOP model that features two techniques: adaptive thresholding (Chen et al., 2020b) and localized context pooling. However, these two types of methods mainly focus on mention- and entity-level information, ignoring the dependencies among entity pairs in same context, which have an important impact on document-level RE.

In contrast, Zhang et al. (2021) and Tan et al. (2022) exploit the dependencies among entity pairs to facilitate document-level RE, however, ignore the fact that relation prediction difficulties of entity pairs are different. Furthermore, Zeng et al. (2021) divide entity pairs into intra- and inter-sentence ones, which are considered to have different prediction difficulties. However, such a division is too simple to accurately distinguish easily- and difficultly-predicted entity pairs. Unlike these methods only utilizing feature-level information of entity pairs in one pass, we leverage both feature- and relation-level information of entity pairs through iterative inference, allowing our model to capture the dependencies among entity pairs more comprehensively.

Besides, our work is inspired by the mask-predict decoding strategy for non-autoregressive NMT (Ghazvininejad et al., 2019; Zhou et al., 2022). In this work, we adapt this strategy into document-level RE. To the best of our knowledge, our work is the first attempt to exploit iterative decoding for this task. Finally, our work is also related to the studies on GNN-based iterative encoding (Lee et al., 2018; Wadden et al., 2019; Lai et al., 2021). Unlike these studies, we train inference module to directly correct the noisy predictions of base module in a one-pass manner rather than iterative manner, while iterative inference is used only during testing.

5 Conclusion and Future Work

In this work, we have proposed a novel document-level RE model with iterative inference, which mainly contains two modules: base module and inference module. We first use base module to make preliminary predictions on the relations of entity pairs. Then, via iterative inference, inference module gradually refines the predictions of base module, which is expected to effectively deal with difficultly-predicted entity pairs depending on other pairs. Experiments on three public document-level RE datasets show that our model significantly outperforms existing competitive baselines. In future, we attempt to apply our model to other inter-sentence or document-level NLP tasks, such as cross-sentence collective event detection.

Acknowledgments

Limitations

The limitations of our method mainly include following two aspects: 1) since our method mainly focuses on relational inference, which is rarely required in sentence-level RE, it has low scalability to sentence-level RE. 2) because relational inference is a complex problem, we require a significant amount of relational inference-specific labeled data to effectively train our model.

References

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. 2019. Matching the blanks: Distributional similarity for relation learning. In *Proceedings of ACL*, pages 2895–2905.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of EMNLP*, pages 3615–3620.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. 2020a. A simple framework for contrastive learning of visual representations. In *Proceedings of ICML*, pages 1597–1607.

Tongfei Chen, Yunmo Chen, and Benjamin Van Durme. 2020b. Hierarchical entity typing via multi-level learning to rank. In *Proceedings of ACL*, pages 8465–8475.

Xinlei Chen and Kaiming He. 2021. Exploring simple siamese representation learning. In *Proceedings of CVPR*, pages 15750–15758.

Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 2019. Connecting the dots: Document-level neural relation extraction with edge-oriented graphs. In *Proceedings of EMNLP*, pages 4925–4936.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL*, pages 4171–4186.

Jun Feng, Minlie Huang, Li Zhao, Yang Yang, and Xiaoyan Zhu. 2018. Reinforcement learning for relation classification from noisy data. In *Proceedings of AAAI*, pages 5779–5786.

Marjan Ghazvininejad, Omer Levy, Yinhan Liu, and Luke Zettlemoyer. 2019. Mask-predict: Parallel decoding of conditional masked language models. In *Proceedings of EMNLP*, pages 6112–6121.

Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. 2017. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*.

Zhijiang Guo, Yan Zhang, and Wei Lu. 2019. Attention guided graph convolutional networks for relation extraction. In *Proceedings of ACL*, pages 241–251.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of CVPR*, pages 9726–9735.

Yuan Huang, Zhixing Li, Wei Deng, Guoyin Wang, and Zhimin Lin. 2021. D-bert: Incorporating dependency-based attention into bert for relation extraction. *CAAI Transactions on Intelligence Technology*, pages 417–425.

Robin Jia, Cliff Wong, and Hoifung Poon. 2019. Document-level n-ary relation extraction with multiscale representation learning. In *Proceedings of NAACL*, pages 3693–3704.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. In *Proceedings of NeurIPS*, pages 18661–18673.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *Proceedings of ICLR*.

Shaopeng Lai, Ante Wang, Fandong Meng, Jie Zhou, Yubin Ge, Jiali Zeng, Junfeng Yao, Degen Huang, and Jinsong Su. 2021. Improving graph-based sentence ordering with iteratively predicted pairwise orderings. In *Proceedings of EMNLP*, pages 2407–2417.

Kenton Lee, Luheng He, and Luke Zettlemoyer. 2018. Higher-order coreference resolution with coarse-to-fine inference. In *Proceedings of NAACL*, pages 687–692.

Bo Li, Wei Ye, Zhonghao Sheng, Rui Xie, Xiangyu Xi, and Shikun Zhang. 2020. Graph enhanced dual attention network for document-level relation extraction. In *Proceedings of COLING*, pages 1551–1560.

Jiao Li, Yueping Sun, Robin J Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Thomas C Wiegers, and Zhiyong Lu. 2016. Biocreative v cdr task corpus: a resource for chemical disease relation extraction. In *Database*.

Ziyang Li, Feng Hu, Chilong Wang, Weibin Deng, and Qinghua Zhang. 2021. Selective kernel networks for weakly supervised relation extraction. *CAAI Transactions on Intelligence Technology*, pages 224–234.

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

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *Proceedings of ICLR*.

Guoshun Nan, Zhijiang Guo, Ivan Sekulic, and Wei Lu. 2020. Reasoning with latent structure refinement for document-level relation extraction. In *Proceedings of ACL*, pages 1546–1557.

Qingyu Tan, Ruidan He, Lidong Bing, and Hwee Tou Ng. 2022. Document-level relation extraction with adaptive focal loss and knowledge distillation. In *Proceedings of ACL Findings*, pages 1672–1681.

Hengzhu Tang, Yanan Cao, Zhenyu Zhang, Jiangxia Cao, Fang Fang, Shi Wang, and Pengfei Yin. 2020. Hin: Hierarchical inference network for document-level relation extraction. In *Proceedings of PAKDD*, pages 197–209.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of NeurIPS*, pages 5998–6008.

Patrick Verga, Emma Strubell, and Andrew McCallum. 2018. Simultaneously self-attending to all mentions for full-abstract biological relation extraction. In *Proceedings of NAACL*, pages 872–884.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, relation, and event extraction with contextualized span representations. In *Proceedings of EMNLP*, pages 5784–5789.

Difeng Wang, Wei Hu, Ermei Cao, and Weijian Sun. 2020a. Global-to-local neural networks for document-level relation extraction. In *Proceedings of EMNLP*, pages 3711–3721.

Hong Wang, Christfried Focke, Rob Sylvester, Nilesh Mishra, and William Wang. 2019a. Fine-tune bert for docred with two-step process. *arXiv preprint arXiv:1909.11898*.

Hong Wang, Christfried Focke, Rob Sylvester, Nilesh Mishra, and William Wang. 2019b. Fine-tune bert for docred with two-step process. *arXiv preprint arXiv:1909.11898*.

Zifeng Wang, Rui Wen, Xi Chen, Shao-Lun Huang, Ningyu Zhang, and Yefeng Zheng. 2020b. Finding influential instances for distantly supervised relation extraction. *arXiv preprint arXiv:2009.09841*.

Tongtong Wu, Xuekai Li, Yuan-Fang Li, Reza Haffari, Guilin Qi, Yujin Zhu, and Guoqiang Xu. 2021a. Curriculum-meta learning for order-robust continual relation extraction. In *Proceedings of AAAI*, pages 10363–10369.

Ye Wu, Ruibang Luo, Henry CM Leung, Hing-Fung Ting, and Tak-Wah Lam. 2019. Renet: A deep learning approach for extracting gene-disease associations from literature. *International Conference on Research in Computational Molecular Biology*, pages 272–284.

Zhiyong Wu, Lingpeng Kong, Wei Bi, Xiang Li, and Ben Kao. 2021b. Good for misconceived reasons: An empirical revisiting on the need for visual context in multimodal machine translation. In *Proceedings of ACL*, pages 6153–6166.

Wang Xu, Kehai Chen, and Tiejun Zhao. 2021. Document-level relation extraction with reconstruction. In *Proceedings of AAAI*, pages 14167–14175.

Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. 2019. DocRED: A large-scale document-level relation extraction dataset. In *Proceedings of ACL*, pages 764–777.

Deming Ye, Yankai Lin, Jiaju Du, Zhenghao Liu, Peng Li, Maosong Sun, and Zhiyuan Liu. 2020. Coreferential Reasoning Learning for Language Representation. In *Proceedings of EMNLP*, pages 7170–7186.

Haiyang Yu, Ningyu Zhang, Shumin Deng, Hongbin Ye, Wei Zhang, and Huajun Chen. 2020. Bridging text and knowledge with multi-prototype embedding for few-shot relational triple extraction. In *Proceedings of COLING*, pages 6399–6410.

Mo Yu, Wenpeng Yin, Kazi Saidul Hasan, Cicero dos Santos, Bing Xiang, and Bowen Zhou. 2017. Improved neural relation detection for knowledge base question answering. In *Proceedings of ACL*, pages 571–581.

Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of ACL*, pages 1753–1762.

Shuang Zeng, Yuting Wu, and Baobao Chang. 2021. SIRE: Separate intra- and inter-sentential reasoning for document-level relation extraction. In *Proceedings of ACL Findings*, pages 524–534.

Shuang Zeng, Runxin Xu, Baobao Chang, and Lei Li. 2020. Double graph based reasoning for document-level relation extraction. In *Proceedings of EMNLP*, pages 1630–1640.

Ningyu Zhang, Xiang Chen, Xin Xie, Shumin Deng, Chuanqi Tan, Mosha Chen, Fei Huang, Luo Si, and Huajun Chen. 2021. Document-level relation extraction as semantic segmentation. In *Proceedings of IJCAI*, pages 3999–4006.

Yuhao Zhang, Peng Qi, and Christopher D. Manning. 2018. Graph convolution over pruned dependency trees improves relation extraction. In *Proceedings of EMNLP*, pages 2205–2215.

Zhenyu Zhang, Bowen Yu, Xiaobo Shu, Tingwen Liu, Hengzhu Tang, Wang Yubin, and Li Guo. 2020. Document-level relation extraction with dual-tier heterogeneous graph. In *Proceedings of COLING*, pages 1630–1641.

Chulun Zhou, Fandong Meng, Jie Zhou, Min Zhang, Hongji Wang, and Jinsong Su. 2022. Confidence based bidirectional global context aware training framework for neural machine translation. In *Proceedings of ACL*, pages 2878–2889.

Huiwei Zhou, Yibin Xu, Weihong Yao, Zhe Liu, Chengkun Lang, and Haibin Jiang. 2020. Global context-enhanced graph convolutional networks for document-level relation extraction. In *Proceedings of COLING*, pages 5259–5270.

Wenxuan Zhou, Kevin Huang, Tengyu Ma, and Jing Huang. 2021. Document-level relation extraction with adaptive thresholding and localized context pooling. In *Proceedings of AAAI*, pages 14612–14620.

Hyper-parameter	DocRED		CDR	GDA
	BERT	RoBERTa-large	SciBERT	SciBERT
At the first stage				
Batch size	8	4	8	6
Epoch number	30	30	20	10
The learning rate for encoder	5e-5	5e-5	1e-5	2e-5
The learning rate for classifier	1e-4	1e-4	5e-5	8e-5
At the second stage				
Batch size	8	4	8	6
Epoch number	15	15	10	5
The learning rate for base module	1e-5	5e-6	1e-6	2e-6
The learning rate for Inference module	1e-4	1e-4	5e-5	8e-5

Table 7: Hyper-parameter Setting.

Figure 6: The case study of our model and our base model (ATLOP). We can observe that our base model usually focuses on identifying the relations of easily-predicted entity pairs. Meanwhile, based on the predictions of base model, our inference module can infer the relations of difficultly-predicted entity pairs. We only show a part of entities within the documents and the according sentences due to the space limitation.

A Hyper-parameters Setting

Table 7 details our hyper-parameter setting. All of our hyper-parameters are tuned on the development set.

B Supplementary Experiments

Case Study Figure 6 shows the case study of our model and our base model (ATLOP). We can observe that ATLOP usually focuses on easily-predicted entity pairs, of which predictions do not require reference to the predicted results of other pairs. Meanwhile, based on the predictions of base model, our inference module can infer difficultly-predicted entity pairs, of which prediction depends on the predicted results of its overlapping pairs.

Ablation studies on CDR and GDA: To further illustrate the effectiveness of different modules in the biomedical field, we also conduct ablation experiments on CDR and GDA datasets. From Table 8, we observe that all the components contribute

Model	CDR	GDA
Ours-SciBERT	**73.2**	**85.9**
w/o ECA	70.5	84.0
w/o fusion sub-layer	72.4	85.2
w/o contrastive loss	71.7	84.6

Table 8: Ablation study of our model on the CDR and GDA test sets.

to the model performance on these two biomedical datasets.