Entity-centered Cross-document Relation Extraction

Fengqi Wang[1], **Fei Li**[1*], **Hao Fei**[2], **Jingye Li**[1], **Shengqiong Wu**[2],
Fangfang Su[1], **Wenxuan Shi**[1], **Donghong Ji**[1] **and Bo Cai**[1*]

[1] Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry
of Education, School of Cyber Science and Engineering, Wuhan University, China
[2] School of Computing, National University of Singapore, Singapore

{wangfengqi,lifei_csnlp,theodorelee,fangsu,shiwenxuan,dhji,caib}@whu.edu.cn
haofei37@nus.edu.sg, swu@u.nus.edu

Abstract

Relation Extraction (RE) is a fundamental task of information extraction, which has attracted a large amount of research attention. Previous studies focus on extracting the relations within a sentence or document, while currently researchers begin to explore cross-document RE. However, current cross-document RE methods directly utilize text snippets surrounding the target entities in multiple given documents, which brings considerable noisy and non-relevant sentences. Moreover, they utilize all the text paths in a document bag in a coarse-grained way, without considering the connections between these text paths. In this paper, we aim to address both of these shortages and push the state-of-the-art for cross-document RE. First, we focus on input construction for our RE model and propose an entity-based document-context filter to retain useful information in the given documents by using the bridge entities in the text paths. Second, we propose a cross-document RE model based on cross-path entity relation attention, which allows the entity relations across text paths to interact with each other. We compare our cross-document RE method with the state-of-the-art methods in the dataset CodRED. Our method outperforms them by at least 10% in F1, thus demonstrating its effectiveness.[1]

1 Introduction

Relation Extraction (RE) aims to detect the semantic relations between a pair of target entities in a given text, which has long been a fundamental task in natural language processing (NLP). Most of RE studies are under the assumption that entity pairs are within a sentence (i.e., sentence-level RE) (Zeng et al., 2014; dos Santos et al., 2015; Cai et al., 2016; Zhou et al., 2016; Zhang et al., 2018; Fei et al., 2021e) or a document (i.e., document-level RE) (Christopoulou et al., 2019; Nan et al.,

2020; Zeng et al., 2020; Li et al., 2021; Fei et al., 2022a; Zhang et al., 2021b). Another line considers the research of cross-text RE, where entity pairs are separated into different text units, (i.e., cross-sentence RE or N-ary RE) (Peng et al., 2017).

The latest RE research has moved to cross-document RE (CodRE), i.e., the target entities are located in different documents (Yao et al., 2021). As exemplified in Fig. 1, a CodRE model needs to first retrieve the relevant documents and then recognizes the key **text paths** in these documents for relation reasoning. In Yao et al. (2021), the task is formalized based on the idea of distant supervision (Mintz et al., 2009), i.e., the text paths in a bag can facilitate the relation reasoning and thus their model performs bag-level prediction over all the text paths. Unfortunately, their method may suffer from at least two problems, which inevitably hinder the accurate relation inference.

First, the inputs of their method are not tailor-made for cross-document RE. For instance, they extract text snippets surrounding the two target entities in the document as input of a bag, brings much noisy and non-relevant context information. Moreover, they ignore important bridge entities in the text paths of the bag, leading to the loss of instructive and salience information for cross-document RE. As can be seen in Fig. 1, the sentences containing bridge entities are necessary to reason the relations between target entities and missing them will seriously affect the reasoning process.

Second, their method does not make full use of the connections between text paths. For example, the pipeline model proposed by Yao et al. (2021) simply leverages the information of the text path in an isolated way, lacking deep consideration of the global connections of all text paths. In contrast, although their end-to-end model (Yao et al., 2021) uses the context of all the text paths, the process of synthesizing the context is coarse-grained. The connections across multiple text paths are actually
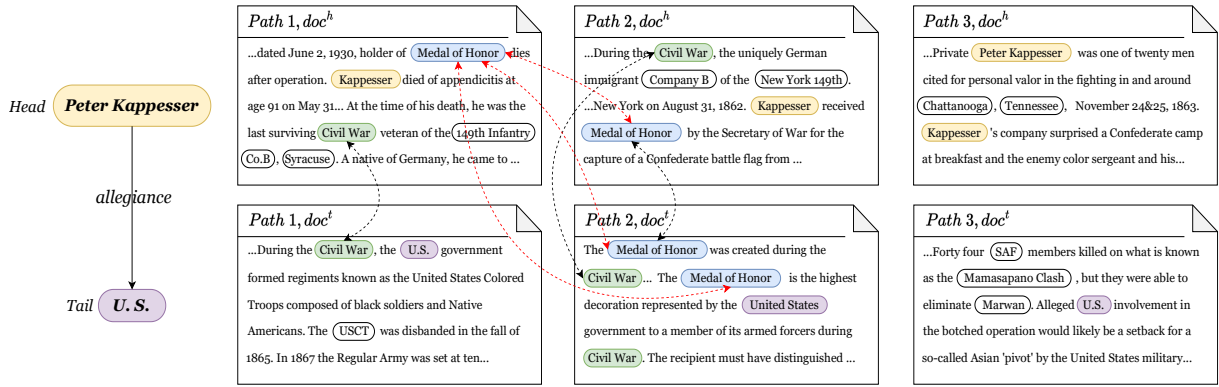

Figure 1: An example to show the setting of cross-document RE. In this document bag, there are three text paths to imply the *allegiance* relation between the head entity *Peter Kappesser* and tail entity *U.S.*. Each text path has two documents, where one contains the head entity and the other one contains the tail entity. In each text path, the head and tail entities are bridged by another entity appearing in both documents (e.g., *Civil War*).

beneficial for cross-document RE. As shown in Fig. 1, the entity "*Medal of Hornor*" provides an additional link for different text paths, which helps to reason the "*allegiance*" relation between "*Peter Kappesser*" and "*U.S.*".

Therefore, in this paper, we focus on addressing the above problems and improving the performance of cross-document RE by presenting a novel **E**ntity-based **C**ross-path **R**elation **I**nference **M**ethod (**ECRIM**). First, we propose an **entity-based document-context filter** to elaborately construct the input for our cross-document RE model, which includes two steps: **1)** We filter out a number of sentences based on their scores with regards to bridge entities. Three heuristic conditions are used to describe the importance scores of bridge entities and then these scores are assigned to the sentences for filtering. **2)** After filtering out the sentences with lower scores, we use the semantic-based sentence filter to reorder the remaining sentences, making them into a relatively coherent document, inspired by the method of sentence ordering in multi-document summarization. (Barzilay and Elhadad, 2011; Ekmekci et al., 2019).

After input construction, we propose a novel cross-document RE model that is equipped with a **cross-path entity relation attention** module to capture the connections of text paths within a document bag, inspired by Zhou et al. (2009); Tu et al. (2019). Specifically, we build a relation matrix where each unit represents a relation between two entities belonging to the same bag. Then the bag-level relation matrix is able to capture the dependencies between the relations by the attention mechanism (Vaswani et al., 2017), which allows one relation to focus on other more relevant relations in

the text paths by modeling the discourse structure (Fei et al., 2022b, 2020a; Wu et al., 2022).

We conduct experiments on the CodRED dataset (Yao et al., 2021). The results show that our model outperforms the baseline models by a large margin. In summary, our contributions can be summarized as follows:

- We apply an entity-based document-context filter to retain useful context information and important bridge entities across the documents.
- We propose a cross-path entity relation attention model for cross-document RE, which allows the relation representations across text paths to interact with each other respect to bridge entities.
- We validate the effectiveness of our model, which significantly pushes the state-of-the-art performance for cross-document RE.

2 Related Work

2.1 Sentence-level Relation Extraction

Relation Extraction is one of the key tasks of information extraction community (Ren et al., 2018; Fei et al., 2020b, 2021b,a; Cao et al., 2022; Li et al., 2022). Sentence-level RE aims at identifying the relationship between two entities in a sentence and many efforts have been devoted to this problem. Zeng et al. (2014) exploit a convolutional deep neural network to extract lexical and sentence-level features. dos Santos et al. (2015) tackle the sentence-level RE task by using a convolutional neural network that performs classification by ranking. Cai et al. (2016) present a novel model that
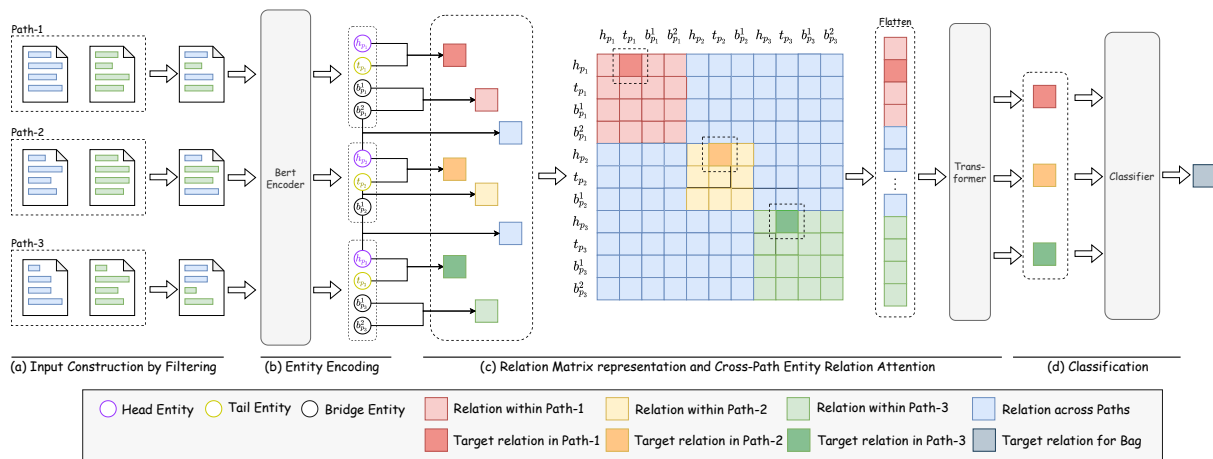
Figure 2: The overall architecture of our system. (a) utilizes a entity-based document-context filter to select the sentences that are relevant to the target entity pair (cf. Section 3.1). (b) yields entity embeddings from contextualized word representations (cf. Section 3.2). (c) leverages the cross-path entity relation attention to capture the connections between the entities and relations of all the paths in the bag (cf. Section 3.3). (d) aggregates the predictions of all the paths to get a bag-level prediction.

strengthen the ability to classifying directions of relationships between entities. Zhang et al. (2018) propose an extension of graph convolutional networks (Wei et al., 2019, 2020) and applied a novel pruning strategy to incorporate relevant information while removing irrelevant content.

2.2 Document-level Relation Extraction

Recent years, researchers have shown a growing interest for document-level text mining (Fei et al., 2021c,d; Zhang et al., 2021a; Yang et al., 2021). Document-level RE aims to detect the relations within one document. Christopoulou et al. (2019) utilize different types of nodes and edges to create a document-level graph for document-level RE. Nan et al. (2020) propose a novel model that empowers the relational reasoning across sentences by automatically inducing the latent document-level graph. Zeng et al. (2020) propose a graph aggregation and inference network to infer relations between entities across long paragraphs. Li et al. (2021) devise a novel mention-based reasoning module based on explicitly and collaboratively local and global reasoning. Zhang et al. (2021b) regard document-level RE as a semantic segmentation task and developed a document U-shaped network to capture both local context information and global interdependency among triples for document-level RE.

2.3 Cross-Document Relation Extraction

Earlier, some researchers probe into extracting entities, events, and relations from text in cross-document setting(Zaraket and Makhlouta, 2012;

Makhlouta et al., 2012). Recently, Cross-document Relation Extraction has been explored deeply by Yao et al. (2021), who presents the first large-scale CodRE dataset, CodRED. To accomplish the task, Yao et al. (2021) propose two solutions, including a pipeline model and a joint model. The pipeline method first extracts a relational graph for each document, and then reasons over these graphs to extract the target relation; while the joint method directly aggregates different text path representations via a selective attention mechanism for the relation prediction. We note that an effective CodRE system requires cross-document multi-hop reasoning through multiple potential bridging entities to narrow the semantic gap between documents. However, the best-performing joint model in Yao et al. (2021) suffers from coarse-grained reasoning by merely synthesizing text paths in a shallow manner. In this work, we consider modeling the global dependencies across multiple text paths (i.e., crosspath) based on bridging entities, which ensures more reliable reasoning for CodRE.

3 Framework

Task Definition Given a target entity pair (e^h, e^t) and a bag of N text paths $B = \{p_i\}_{i=1}^N$, where each path p_i consists of two documents (d_i^h, d_i^t) mentioning the head entity e^h and the tail entity e^t separately, the task aims to infer the relation r from $\mathcal{R}$ between the target entity pair, where $\mathcal{R}$ is a pre-defined relation type set. When multiple mentions of one entity (subject to entity ID) appear in two documents respectively, this entity is said
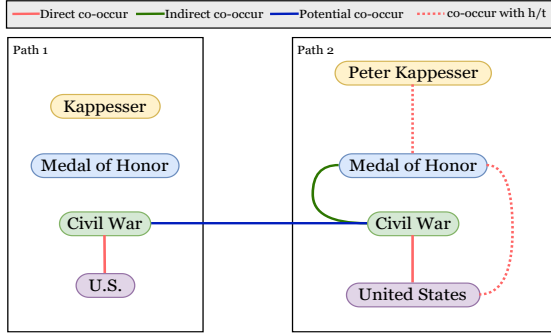
Figure 3: An example of the co-occurring graph for Path 1 and Path 2 in Fig.1. The score of "*Civil War*" is obtained by aggregating the scores obtained from three conditions $\Gamma_1, \Gamma_2, \Gamma_3$ as shown in Equation (1).

to be shared by two documents. Note that the two documents in every path may share multiple entities $E_i^b = \{e_i^b\}_{i=1}^M$, in the following we call them bridge entities.

System Overview As shown in Fig.2, the model consists of four tiers. First, an entity-based document-context filter receives text paths as inputs, where each of them is composed of two documents. The filter removes less relevant sentences from the text paths and reorganizes the remaining sentences into more compact inputs for subsequent tiers. Afterward, a BERT encoder yields the representations for tokens and entities. Then the cross-path entity relation attention module builds a bag-level entity relation matrix for capturing the global dependencies between the entities and relations in the bag, and outputs the entity relation representations of all text paths. Finally, we use a classifier to aggregate these representations and predict the relation between head and tail entities.

3.1 Entity-based Document-context Filter

Since the average length of a document in CodRED is more than 4,900 tokens and BERT has a length limitation (512 tokens) for input, it is infeasible to handle all sentences in a text path simultaneously if the total length of all the input exceeds the limitation. To solve this problem, we propose an entity-based document-context filter to select salient sentences in a document for each path.

For each path p, we have a collection of entities E^b shared by the two documents (d^h, d^t) of this text path. These bridge entities can be utilized as a link in reasoning about the relation between head/tail entities. Moreover, the bridge entity collections can be regarded as a latent indicator to measure the distribution similarity between differ-

ent text paths. Thus, we first filter out a number of sentences based on their scores, which are computed by three heuristic conditions. Then we use a semantic-based sentence filter to reorder the selected sentences to construct a coherent document whose length is less than 512.

3.1.1 Entity-based Sentence Filtering

The basic assumption of this module is that *If a sentence includes entities that co-occur with a target entity, the sentence is informative for relation reasoning*. Thus our first filtering procedure is to select those informative sentences with prior distribution knowledge of bridge entities. To this end, we use three steps:

Step 1: We calculate the co-occurring score for each bridge entity. We design three heuristic conditions from strong to weak to describe the different levels of co-occurring situations:

- **Direct co-occur** (Γ_1): Whether it co-occurs with the head/tail entity in the same sentence.
- **Indirect co-occur** (Γ_2): Whether it co-occurs with another entity meets the first condition.
- **Potential co-occur** (Γ_3): Whether it exists in other text paths.

Formally, for a bag of N text paths, we score for each bridge entity e^b in each text path p_i by:

$$\text{score}(e^b) = \alpha s_1(e^b) + \beta s_2(e^b) + \gamma s_3(e^b) \quad (1)$$

$$s_1(e^b) = \begin{cases} 1, & \text{if } \Gamma_1(e^b) \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

$$s_2(e^b) = \begin{cases} |\{e^o|\Gamma_1(e^o) \wedge I(e^o) = 1\}|, & \text{if } \Gamma_2(e^b) \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

$$s_3(e^b) = \begin{cases} \left|\left\{p_j | e^b \in E_j^b\right\}\right|, & \text{if } \Gamma_3(e^b) \\ 0, & \text{otherwise} \end{cases} \quad (4)$$

where α, β, γ are hyper-parameters. $I(e^o) = 1$ while e^o and e^b co-occur in the same sentence, where $e^o \in E_i^b \setminus \{e^b\}$. equation(3) sums number of these e^o, equation(4) sums number of these p_j.

Step 2: We compute the importance score g^s of each sentence s by summarizing all the scores of the bridge entities that it contains:

$$g^s = \sum_{e^b \in E_s^b} \text{score}(e^b) \quad (5)$$

where E_s^b denotes the bridge entities mentioned in the sentence s.

Step 3: We rank the sentences by their importance scores from large to small and select the top K sentences as the candidate set $S = \{s_1, s_2, ..., s_K\}$, where K is a hyper-parameter. In our implementation, the candidate set size K is set to 16 based on the experiments on the development set. If there are several sentences with the same score, the priority is determined according to the distances from these sentences to the sentence with the highest score.

3.1.2 Semantic-based Sentence Filtering

After the entity-based sentence filtering, we take the semantic relevance of sentences into account to further filter and reorder candidate sentences, with the assumption that *if a sentence is semantically similar to the sentence including target entities, this sentence should be more informative for relation reasoning*. The goal of this step is to yield the most informative context S^* from the candidate sentence set S, for reasoning the relation between target entities.

The procedure of semantic-based sentence filtering is summarized as Algorithm 1, which aims to construct the sequence S^* from the candidate sentence set. As seen, besides the candidate set S, head entity h and tail entity t, the inputs of the algorithm also include a start set S_{start} and an end set S_{end} that consist of all the sentences containing the head and tail entity, respectively. At the begging of the algorithm, we first randomly select a sentence from S_{start} (line 1). Then we search for the most relevant sentence to this sentence and append it to the output S^*. We repeat such a process until the current selected sentence includes the tail entity (lines 3-12). Finally, we obtain the sequence S^* with K^* sentences, where $K^* \leq K$. Specifically, we use the cosine similarity calculated by SBERT-WK (Wang and Kuo, 2020) to measure the semantic relevance between two sentences. If the length of the sequence S^* is larger than 512, we will keep dropping the sentences with lower similarity scores until the length of the sequence meets the demand of BERT.

3.2 Encoder Module

After input construction, we have filtered sentence set S^* from each text path, we concatenate sentences in S^* together to build the input of our model as $X = \{w_i\}_{i=1}^L$. Following Yao et al. (2021), we apply unused tokens in the BERT vocabulary (Devlin et al., 2019) to mark the start and end of every

---

Algorithm 1 Semantic-based Sentence Filtering

Input: Candidate set $S = \{s_1, s_2, ..., s_K\}$; head entity h; tail entity t; Start set $S_{start} = \{s_i | h \in s_i\}$; End set $S_{end} = \{s_j | t \in s_j\}$;
Output: Sequential sentences S^*
1:   $S^* = [\ \ ]; cur = \text{Random}(S_{start}, 1)$
2:   $next = \phi; max = 0$
3: **while** $\{cur\} \cap S_{end} = \emptyset$ **do**
4:      $S = S - \{cur\}$
5:      **for** $s_i \in S$ **do**
6:         **if** $\text{Sim}(cur, s_i) > max$
7:           $max \leftarrow \text{Sim}(cur, s_i)$
8:           $next \leftarrow s_i$
9:         **else**
10:          continue
11:      $S^* = \text{Append}(S^*, next)$
12:      $cur \leftarrow next$
13: **return** S^*

---

entity. Then we leverage BERT as the encoder to yield token representations:

$$\{\boldsymbol{h}_i\}_{i=1}^L = \text{BERT}(\{w_i\}_{i=1}^L) \quad (6)$$

Based on $\{\boldsymbol{h}_i\}_{i=1}^L$, we can obtain the entity representations with the max-pooling operation:

$$\boldsymbol{e}_j = \text{Max}\,\{\boldsymbol{h}_i\}_{j=start_j}^{end_j} \quad (7)$$

where $start_j$ and end_j are the start and end positions of the j-th mention.

3.3 Cross-Path Entity Relation Attention

Since prior studies only treated each text path as an independent instance, the rich information across text paths was ignored. Therefore, we aim to mine this information. Inspired by Jin et al. (2020) and Zhang et al. (2021b), we introduce a cross-path entity relation attention module based on the Transformer (Vaswani et al., 2017) to capture the interdependencies among the relations across paths.

Concretely, we first collect all the entity mention representations in a bag and then generate relation representations for entity pairs:

$$\boldsymbol{r}_{u,v} = \text{ReLU}(\boldsymbol{W}_r(\text{ReLU}(\boldsymbol{W}_u\boldsymbol{e}_u + \boldsymbol{W}_v\boldsymbol{e}_v))) \quad (8)$$

where $\boldsymbol{W}_r$, $\boldsymbol{W}_u$, $\boldsymbol{W}_v$ are learnable parameters. Afterward, we extend the relation matrix proposed by Jin et al. (2020) at the bag level, as shown in Fig. 2(c). In order to modeling the interaction among relations across paths, we build a relation matrix $\boldsymbol{M} \in \mathbb{R}^{|E| \times |E| \times d}$, where $E = \bigcup_{i=1}^N E_i$ denotes all the entities in the entity set E_i of text path p_i and $E_i = \{e_i^h, e_i^t\} \cup E_i^b$.

To capture the intra- and inter-path dependencies, we leverage a multi-layer Transformer (Vaswani et al., 2017) to perform self-attention on the flattened relation matrix $\hat{\boldsymbol{M}} \in \mathbb{R}^{|E|^2 \times d}$:

$$\hat{\boldsymbol{M}}^{(t+1)} = \text{Transformer}(\hat{\boldsymbol{M}}^{(t)}) \qquad (9)$$

Finally, we obtain the target relation representation $\boldsymbol{r}_{h_i,t_i}$ for each path p_i from the last layer of the Transformer, as shown in Figure 2(c).

3.4 Classifier

Afterwards, we yield the relation representation $\boldsymbol{r}_{h_i,t_i}$ from each text path p_i for each pair of target entities. Then we use the $\boldsymbol{r}_{h_i,t_i}$ as the classification feature and feed it into an MLP classifier for calculating the score of each relation:

$$\hat{y}_i = \text{MLP}(\boldsymbol{r}_{h_i,t_i}) \qquad (10)$$

To get the bag level prediction, we use the max-pooling operation on each relation label to yield the final score for each relation type r:

$$\hat{y}^{(r)} = \text{Max}\left\{ \hat{y}_i^{(r)} \right\}_{i=1}^N \qquad (11)$$

After obtaining the scores for all relations, we utilize a global threshold θ, which will be stated in Section 3.5, to filter out the categories lower than the threshold.

3.5 Training Details

Since some bags have multiple relation labels, we adopt a multi-label global-threshold loss, which is a variant of the circle loss (Sun et al., 2020), as our loss function. To this end, we introduce an additional threshold to control which class should be output. We hope that the scores of the target classes are greater than the threshold and the scores of the non-target classes are less than the threshold. Formally, for each Bag B, we have:

$$\begin{aligned}
\mathcal{L} = {} & log(e^{\theta} + \sum_{r \in \Omega_{neg}^B} e^{\hat{\boldsymbol{y}}^{(r)}}) \\
& + log(e^{-\theta} + \sum_{r \in \Omega_{pos}^B} e^{\hat{\boldsymbol{y}}^{(r)}})
\end{aligned} \qquad (12)$$

where $\hat{\boldsymbol{y}}^{(r)}$ denotes the score for the relation r, θ denotes the threshold and is set to zero, Ω_{pos}^B and Ω_{neg}^B are the positive and negative classes between the target entity pair.

4 Experimental Settings

4.1 CodRED Dataset

The CodRED dataset was constructed by Yao et al. (2021) from the English Wikipedia and Wikidata,

	Train	Dev	Test
Bags (Pos)	2,733	1,010	1,012
Bags (N/A)	16,668	4,558	4,523
Text paths	129,548	40,740	40,524
Bridges	613,566	195,766	197,888
Tokens/Doc	4,938.6	5,031.6	5,129.2
Path/Bag	6.67	7.31	7.32

Table 1: Statistics of CodRED.

which covers 276 relation types. The statistics of our data are shown in Table 1, which is the same as that used in Yao et al. (2021).

4.2 Implementation Details and Evaluation Metrics

We conduct our experiments using the closed setting of the benchmark dataset CodRED.[2] We use the cased BERT-base as the encoder. AdamW (Loshchilov and Hutter, 2019) is used to optimize the neural networks with a linear warm-up and decay learning rate schedule. The learning rate is 3e-5, and the embedding and hidden dimension is 768. The α, β, γ in 3.1.1 are 0.1, 0.01, 0.001 respectively. The Transformer encoder in 3.3 have 3 layers. We tuned the hyper-parameters on the development set. Other parameters in the network are all obtained by random initialization and updated during training.

Following Yao et al. (2021), we adopt the F1/AUC/P@500/P@1000 (ignore N/A predictions) as the evaluation metrics for the experiments on the development set, and F1/AUC (ignore N/A predictions) for the experiments on the test set. Results are obtained from CodaLab.[3] For each target entity pair, the model yield a logit for each relation type. We rank (h, t, r) according to the logit values from high to low, and select the top-N values to compute an average precision called P@N. The F1/AUC/P@N (ignore N/A predictions) means that logits of $(h, t, n/a)$ will not be included in the calculation of F1/AUC/P@N.

4.3 Baselines

We compare our proposed model with two baselines provided by Yao et al. (2021).

Pipeline. Yao et al. (2021) build a pipeline model that decomposes cross-document RE into three phases: 1) firstly, predicting the relations between the entities within a document to yield a relational graph containing head or tail entities; 2) secondly,

---

[2]Available at https://github.com/thunlp/CodRED.
[3]https://codalab.lisn.upsaclay.fr/competitions/3770.

Model	Dev				Test	
	F1	AUC	P@500	P@1000	F1	AUC
Pipeline (Yao et al., 2021)	30.54	17.45	30.60	26.70	32.29	18.94
End-to-end (Yao et al., 2021)	51.26	47.94	62.80	51.00	51.02	47.46
ECRIM (ours)	**61.12**	**60.91**	**78.89**	**60.17**	**62.48**	**60.67**

Table 2: Comparisons with the baselines on CodRED. The results of the baselines are extracted from the original paper. Our test results are obtained from the official website of CodRED on Codalab.

for each entity e shared by two relational graphs, predicting the relation (h, e) and (e, t) respectively, then concatenate the two relation representation and feed it into a fully connected layer to obtain relation distribution; 3) finally, aggregating the relation scores for all shared entity e to obtain the final relation between the target entity pair.

End-to-end. Yao et al. (2021) also design an end-to-end model to predict the relation. Specifically, they obtain representation for each text path p_i by feeding tokens into BERT. Then they use selective attention mechanism to obtain an aggregated representation from all paths. Finally the aggregated representation is fed into a fully connected layer followed by a softmax layer to predict the relation between the entity pair.

5 Results and Analyses

5.1 Main Results

In this section, we report the main experimental results compared with the baseline models proposed by Yao et al. (2021). As shown in table 2, our model achieves superior performance in all metrics for both development set and test set. Specially, our method achieves 62.48% F1 and 60.67% AUC on the test set, and outperforms the best method End-to-end by 11.46% and 13.21% in terms of F1 and AUC. The improvement in those scores verifies the excellent ability of our model due to our design for bridge entities and cross-path interaction. These two points will be further discussed in 5.3 and 5.4.

5.2 Ablation Studies

In this section, we conduct ablation experiments to verify the effectiveness of each component of our model. We implement following model variants:

(1) $\text{ECRIM}_{w/o\ IC}$, a variant that replaces the input construction module with the method used by Yao et al. (2021), which evaluates the contribution of the input construction module.

(2) $\text{ECRIM}_{w/o\ BR}$, a variant that discards bridge entities when constructing relation matrix, i.e. the relation matrix merely composed of the relations of target entities.

Model	F1	AUC	P@500	P@1000
ECRIM	61.12	60.91	78.89	60.17
-IC	59.14	59.16	75.78	58.46
-BR	56.99	57.85	73.21	56.80
-CP	58.63	59.27	74.36	57.62
-TH	60.22	59.80	75.42	58.74

Table 3: Ablation studies on the CodRED development set. IC: Input Construction; BR: Bridge Entity; CP: Cross-Path Entity Relation Attention; TH: Threshold.

(3) $\text{ECRIM}_{w/o\ CP}$, a variant that uses inner-path entity relation attention instead of cross-path entity relation attention, which is to validate the effectiveness of the utility of relation dependencies flow across text paths.

(4) $\text{ECRIM}_{w/o\ TH}$, a variant that replaces the threshold loss with the cross entropy loss.

All these variants use the BERT-based as encoder. And the results are presented in Table 3, from which we can observe that:

(1) The performance of $\text{ECRIM}_{w/o\ IC}$ drops significantly, which confirms the importance of retaining significant information related to bridge entities.

(2) The result of $\text{ECRIM}_{w/o\ BR}$ variant shows that when bridge entities were ablated, the performance of model declined substantially. This proves that relations with respect to bridge entities are very important.

(3) The performance of $\text{ECRIM}_{w/o\ CP}$ model drops significantly as the cross-path entity relation attention module is discarded and replaced with Inner-Path Entity Relation Attention. This phenomenon indicates the effectiveness of enabling relations to interact with each other across text paths.

(4) The performance of $\text{ECRIM}_{w/o\ TH}$ variant has decreased, demonstrating the effectiveness of the threshold loss we used.

5.3 Effect on the Number of Bridge Entities

To investigate the effect of bridge entities for cross-document relation extraction, we divided the origin dev set of CodRED into several subsets by the av-
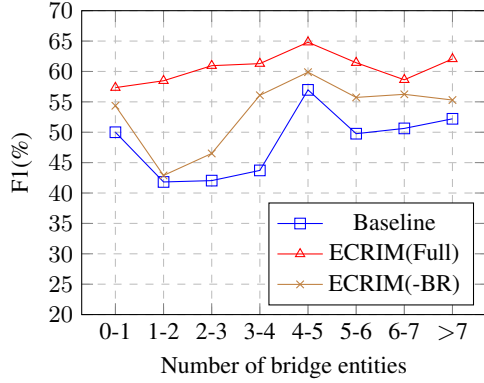
Figure 4: The effect on F1s with regards to different numbers of bridge entities per path in bags.
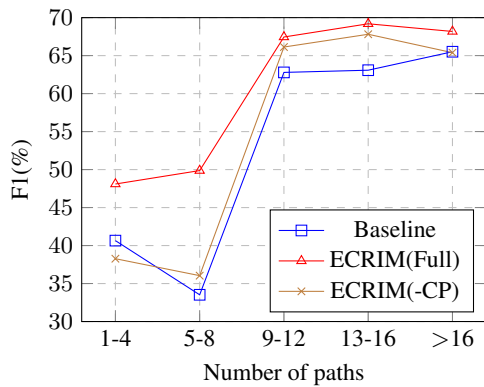

Figure 5: The effect on F1s with regards to different numbers of paths in bags.

Figure 6: The effect on F1s and AUCs with offset under the different selections of K in Section 3.1.

	Speed(bags $\cdot$ min^{-1})
$K = 8$	37.89
$K = 12$	15.43
$K = 16$	8.28
$K = 20$	4.72
$K = 24$	2.53

Table 4: Comparison of algorithm 1 execution speed under different K value settings.

erage number of bridges per path in a bag. We report the model performance on these subsets as shown in Fig.4. We can observe that as the number of bridge entities increases, the performance of the model increases first because the bridge entities bring more information shared by the two documents in the path. This evidently proves the necessity to utilize the bridge entity information for cross-document RE.

As the number of bridge entities in the path continues to increase, the performance of the model decreases slightly. This is due to the increase in noise caused by a large number of bridge entities. The complex context makes the reasoning process of the model difficult. However, our model is better than baseline in resisting this noise, as our model can distinguish noise factors in finer granularity.

5.4 Effect Analysis for the Number of Paths

To investigate the impact of path numbers within a bag for CodRE, we divided the origin dev set of CodRED into several subsets by path numbers of each bag. We report the model performance on
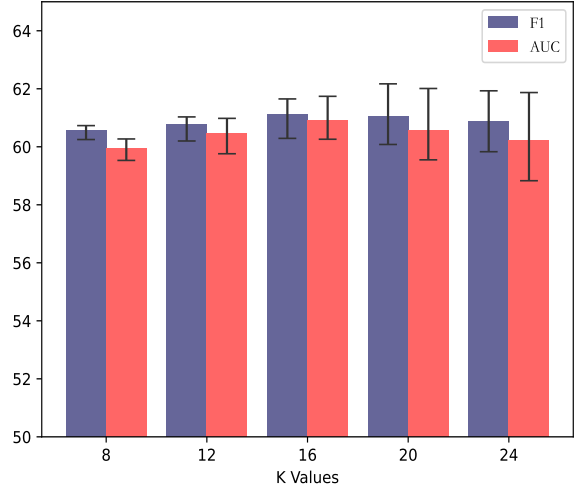
these subsets as shown in Fig.5. We can observe that all models perform better with a larger number of paths than with a small number of paths, as the number of positive paths in a bag also increases. Our model ECRIM (Full) achieves a great improvement compared to baseline when the number of paths is small, which shows that when facing difficult situations with fewer paths, our model makes full use of cross-path information for reasoning.

5.5 Which is the appropriate value for hyperparameter K?

In this section, we experiment on the development set to heuristically search for the appropriate value of hyperparameter K. The influence of K value is mainly in two aspects: constraint for the uncertainty brought by algorithm 1 and constraint the computational time cost for the execution of algorithm 1. Figure.6. shows that with the increase of K value, the fluctuation degree of model performance is greater. On the other hand, table 4 shows the significant impact of K value on the algorithm execution time. Considering the effects of both aspects, we set the value of K to 16 to ensure that the fluctuation is small when the time cost is acceptable.

MBE	MUA	TMU	ST
2	1940	10859	5.03
3	2580	11499	5.06
4	3630	12549	5.01
5	5256	14175	5.03
6	7822	16741	4.93
7	11418	20337	4.87
8	15334	24253	4.67
>8	-	Out of Memory	-

Table 5: GPU memory usage and running speed for different number of bridge entities. MBE denotes Max number of Bridge Entities per path, MUA denotes Memory Usage (MiB) of Attention matrix, TMU denotes Total Memory Usage (MiB) of Model and ST represents the Speed ($bags \cdot min^{-1}$) on the Train set.

5.6 Case Study

To further illustrate the effectiveness of cross-path dependency between relations learned by our model, we present a case study which can be seen in Fig.7, an attention score matrix heatmap of relation (h_1, t_1). For example, the unit in row 3 and column 2 (whose coordinate is $(t(1), b(1)(1))$) represents the contribution of the relation between the tail entity of p_1 and the first bridge entity of $p1$ to r_{h_1,t_1}, where $t(i)$ represents the tail entity of the i-th path, and $b(i)(j)$ denotes the j-th bridge entity of the i-th path. Obviously, the most prominent areas on the heatmap are the four blocks in the upper left corner, which denote the inner relation of p_1 and p_2 and the cross path relation between p_1, p_2. As the ground-truth label of these paths is $[P126, P126, n/a, n/a, n/a, n/a]$ ($P126$ and n/a refers to *relation ID* and *no relation*), it is proved that the model successfully learns the cross-path dependency which contributes to the prediction.

6 Conclusions

In this paper, we devise an entity-based document-context filter to extract important snippets related to the target information for cross-document RE. For relation prediction, we propose a model that considers the global dependencies across multiple text paths and performs a fine-grained reasoning process simultaneously. Empirical results show that our method drastically improves the performance of cross-document relation extraction. Our work can be a valuable reference for this research.

Limitations

Because the model is built at the bag level, the computational complexity of cross-path entity relation
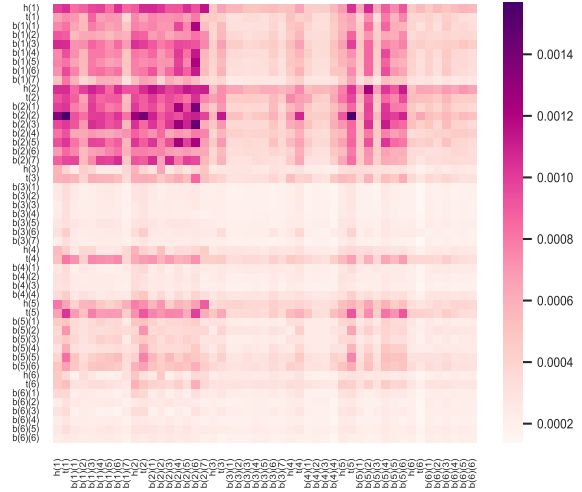

Figure 7: A case study to show the normalized attention scores of a target relation unit (h_1, t_1) for all the relations in the bag.

attention will grow with the increasing numbers of text paths and bridge entities, resulting in an increase in GPU memory demand and a decrease in inference speed. As table 5 demonstrated, the usage of GPU memory increases rapidly with the increase of bridge entity number, as the shape of the Attention Matrix is $|E|^4 \times d$. Meanwhile, the computing efficiency decrease slightly. If there are too many potential paths, we have to discard some of them to maintain the feasibility of our model.

In addition, the entity-based document-context filter that we use to construct the input is unsupervised and not learnable. How to build a learnable model to extract more informative sentences from long documents is a future work that has much room for exploration. Another potential line is to explicitly model the discourse structure of the relevant documents, over which the reasoning of the RE or cross-document RE will be easier.

References

Regina Barzilay and Noemie Elhadad. 2011. Inferring strategies for sentence ordering in multidocument news summarization. *CoRR*, abs/1106.1820.

Rui Cai, Xiaodong Zhang, and Houfeng Wang. 2016. Bidirectional recurrent convolutional neural network for relation classification. In *Proceedings of the ACL*, pages 756–765.

Hu Cao, Jingye Li, Fangfang Su, Fei Li, Hao Fei, Shengqiong Wu, Bobo Li, Liang Zhao, and Donghong Ji. 2022. OneEE: A one-stage framework for fast overlapping and nested event extraction. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1953–1964.

Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 2019. Connecting the dots: Document-level neural relation extraction with edge-oriented graphs. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (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 the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186.

Cícero dos Santos, Bing Xiang, and Bowen Zhou. 2015. Classifying relations by ranking with convolutional neural networks. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 626–634.

Berk Ekmekci, Eleanor Hagerman, and Blake Howald. 2019. Specificity-based sentence ordering for multi-document extractive risk summarization. *CoRR*, abs/1909.10393.

Hao Fei, Donghong Ji, Bobo Li, Yijiang Liu, Yafeng Ren, and Fei Li. 2021a. Rethinking boundaries: End-to-end recognition of discontinuous mentions with pointer networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 12785–12793.

Hao Fei, Fei Li, Bobo Li, and Donghong Ji. 2021b. Encoder-decoder based unified semantic role labeling with label-aware syntax. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 12794–12802.

Hao Fei, Jingye Li, Shengqiong Wu, Chenliang Li, Donghong Ji, and Fei Li. 2022a. Global inference with explicit syntactic and discourse structures for dialogue-level relation extraction. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI*, pages 4082–4088.

Hao Fei, Yafeng Ren, and Donghong Ji. 2020a. Mimic and conquer: Heterogeneous tree structure distillation for syntactic NLP. In *Findings of the Association*

for Computational Linguistics: EMNLP 2020, pages 183–193.

Hao Fei, Yafeng Ren, Shengqiong Wu, Bobo Li, and Donghong Ji. 2021c. Latent target-opinion as prior for document-level sentiment classification: A variational approach from fine-grained perspective. In *Proceedings of the WWW: the Web Conference*, pages 553–564.

Hao Fei, Shengqiong Wu, Yafeng Ren, Fei Li, and Donghong Ji. 2021d. Better combine them together! integrating syntactic constituency and dependency representations for semantic role labeling. In *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021*, pages 549–559.

Hao Fei, Shengqiong Wu, Yafeng Ren, and Meishan Zhang. 2022b. Matching structure for dual learning. In *Proceedings of the International Conference on Machine Learning, ICML*, pages 6373–6391.

Hao Fei, Meishan Zhang, and Donghong Ji. 2020b. Cross-lingual semantic role labeling with high-quality translated training corpus. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7014–7026.

Hao Fei, Yue Zhang, Yafeng Ren, and Donghong Ji. 2021e. A span-graph neural model for overlapping entity relation extraction in biomedical texts. *Bioinformatics*, 37(11):1581–1589.

Zhijing Jin, Yongyi Yang, Xipeng Qiu, and Zheng Zhang. 2020. Relation of the relations: A new paradigm of the relation extraction problem. *CoRR*, abs/2006.03719.

Jingye Li, Hao Fei, Jiang Liu, Shengqiong Wu, Meishan Zhang, Chong Teng, Donghong Ji, and Fei Li. 2022. Unified named entity recognition as word-word relation classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10965–10973.

Jingye Li, Kang Xu, Fei Li, Hao Fei, Yafeng Ren, and Donghong Ji. 2021. MRN: A locally and globally mention-based reasoning network for document-level relation extraction. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1359–1370.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Jad Makhlouta, Fadi Zaraket, and Hamza Harkous. 2012. Arabic entity graph extraction using morphology, finite state machines, and graph transformations. In *Computational Linguistics and Intelligent Text Processing*, pages 297–310.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011.

Guoshun Nan, Zhijiang Guo, Ivan Sekulic, and Wei Lu. 2020. Reasoning with latent structure refinement for document-level relation extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1546–1557.

Nanyun Peng, Hoifung Poon, Chris Quirk, Kristina Toutanova, and Wen-tau Yih. 2017. Cross-sentence n-ary relation extraction with graph lstms. *Trans. Assoc. Comput. Linguistics*, 5:101–115.

Yafeng Ren, Hao Fei, and Qiong Peng. 2018. Detecting the scope of negation and speculation in biomedical texts by using recursive neural network. In *Proceedings of the IEEE International Conference on Bioinformatics and Biomedicine*, pages 739–742.

Yifan Sun, Changmao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. 2020. Circle loss: A unified perspective of pair similarity optimization. In *Proceedings of CVPR*, pages 6397–6406.

Ming Tu, Jing Huang, Xiaodong He, and Bowen Zhou. 2019. Multiple instance learning with graph neural networks. *CoRR*, abs/1906.04881.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Bin Wang and C.-C. Jay Kuo. 2020. Sbert-wk: A sentence embedding method by dissecting bert-based word models. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2146–2157.

Yinwei Wei, Xiang Wang, Liqiang Nie, Xiangnan He, and Tat-Seng Chua. 2020. Graph-refined convolutional network for multimedia recommendation with implicit feedback. In *Proceedings of the 28th ACM international conference on multimedia*, pages 3541–3549.

Yinwei Wei, Xiang Wang, Liqiang Nie, Xiangnan He, Richang Hong, and Tat-Seng Chua. 2019. Mmgcn: Multi-modal graph convolution network for personalized recommendation of micro-video. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 1437–1445.

Shengqiong Wu, Hao Fei, Fei Li, Meishan Zhang, Yijiang Liu, Chong Teng, and Donghong Ji. 2022. Mastering the explicit opinion-role interaction: Syntax-aided neural transition system for unified opinion role labeling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 11513–11521.

Hang Yang, Dianbo Sui, Yubo Chen, Kang Liu, Jun Zhao, and Taifeng Wang. 2021. Document-level event extraction via parallel prediction networks. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 6298–6308.

Yuan Yao, Jiaju Du, Yankai Lin, Peng Li, Zhiyuan Liu, Jie Zhou, and Maosong Sun. 2021. CodRED: A cross-document relation extraction dataset for acquiring knowledge in the wild. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4452–4472.

Fadi A. Zaraket and Jad Makhlouta. 2012. Arabic cross-document NLP for the hadith and biography literature. In *Proceedings of the Twenty-Fifth International Florida Artificial Intelligence Research Society Conference*.

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. 2014. Relation classification via convolutional deep neural network. In *Proceedings of International Conference on Computational Linguistics*, pages 2335–2344.

Shuang Zeng, Runxin Xu, Baobao Chang, and Lei Li. 2020. Double graph based reasoning for document-level relation extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1630–1640.

Long Zhang, Tong Zhang, Haibo Zhang, Baosong Yang, Wei Ye, and Shikun Zhang. 2021a. Multi-hop transformer for document-level machine translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3953–3963.

Ningyu Zhang, Xiang Chen, Xin Xie, Shumin Deng, Chuanqi Tan, Mosha Chen, Fei Huang, Luo Si, and Huajun Chen. 2021b. Document-level relation extraction as semantic segmentation. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pages 3999–4006.

Yuhao Zhang, Peng Qi, and Christopher D. Manning. 2018. Graph convolution over pruned dependency trees improves relation extraction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2205–2215.

Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. 2016. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 207–212.

Zhi-Hua Zhou, Yu-Yin Sun, and Yu-Feng Li. 2009. Multi-instance learning by treating instances as non-i.i.d. samples. In *Proceedings of the 26th Annual International Conference on Machine Learning*, page 1249–1256.