Document-Level Relation Extraction with Cross-sentence Reasoning Graph

Hongfei Liu[1], Zhao Kang[1(✉)], Lizong Zhang[1], Ling Tian[1], and Fujun Hua[2]

[1] University of Electronic Science and Technology of China, Chengdu, China
202022081525@std.uestc.edu.cn,
{zkang,l.zhang,lingtian}@uestc.edu.cn
[2] Research and Development Center TROY Information Technology Co. Ltd., Chengdu, China
huafj@troy.cn

Abstract. Relation extraction (RE) has recently moved from the sentence-level to document-level, which requires aggregating document information and using entities and mentions for reasoning. Existing works put entity nodes and mention nodes with similar representations in a document-level graph, whose complex edges may incur redundant information. Furthermore, existing studies only focus on entity-level reasoning paths without considering global interactions among entities cross-sentence. To these ends, we propose a novel document-level RE model with a **GR**aph information **A**ggregation and **C**ross-sentence **R**easoning network (GRACR). Specifically, a simplified document-level graph is constructed to model the semantic information of all mentions and sentences in a document, and an entity-level graph is designed to explore relations of long-distance cross-sentence entity pairs. Experimental results show that GRACR achieves excellent performance on two public datasets of document-level RE. It is especially effective in extracting potential relations of cross-sentence entity pairs. Our code is available at https://github.com/UESTC-LHF/GRACR.

Keywords: Deep learning · Relation extraction · Document-level RE

1 Introduction

Relation extraction (RE) is to identify the semantic relation between a pair of named entities in text. Document-level RE requires the model to extract relations from the document and faces some intractable challenges. Firstly, a document contains multiple sentences, thus relation extraction task needs to deal with more rich and complex semantic information. Secondly, subject and object entities in the same triple may appear in different sentences, and some entities have aliase, which are often named entity mentions. Hence, the information utilized by document-level RE may not come from a single sentence. Thirdly, there may be interactions among different triples. Extracting the relation between two entities from different triples requires reasoning with contextual features. Figure 1 shows an example from DocRED dataset [21]. It is easy to predict intra-sentence relations because the subject and object appear in the same sentence. However, it has a problem in identifying the inter-sentence relation between "Swedish"

> **Johan Gottlieb Gahn**
> [S1] Johan Gottlieb Gahn (19 August 1745 – 8 December 1818) was a Swedish chemist and metallurgist who Discovered manganese In 1774.··· [S7] In 1784, Gahn was elected a member of the Royal Swedish Academy of Sciences. [S8] He also made a managerial career in Swedish mining.
>
> **Triples**:
> <Johan Gottlieb Gahn, country of citizenship, Swedish>
> <Johan Gottlieb Gahn, member of, Royal Swedish Academy>
> <Royal Swedish Academy, country, Swedish>

Fig. 1. An example of document-level RE excerpted from DocRED dataset.

and "Royal Swedish Academy", whose mentions are distributed across different sentences and there exists long-distance dependencies.

[21] proposed DocRED dataset, which contains large-scale human-annotated documents, to promote the development of sentence-level RE to document-level RE. In order to make full use of the complex semantic information of documents, recent works design document-level graph and propose models based on graph neural networks (GNN) [4]. [1] proposed an edge-oriented model that constructs a document-level graph with different types of nodes and edges to obtain a global representation for relation classification. [12] defined the document-level graph as a latent variable and induced it based on structured attention to improve the performance of document-level RE models by optimizing the structure of document-level graph. [17] proposed a model that learns global representations of entities through a document-level graph, and learns local representations of entities based on their contexts. However, these models simply average the embeddings of mentions to obtain entity embeddings and feed them into classifiers to obtain relation labels. Entity and mention nodes share a similar embedding if certain entity has only one mention. Therefore, putting them in the same graph will introduce redundant information and reduce discrimination.

To address above issues, we propose a novel GNN-based document-level RE model with two graphs constructed by semantic information from the document. Our key idea is to build document-level graph and entity-level graph to fully exploit the semantic information of documents and reason about relations between entity pairs across sentences. Specifically, we solve two problems:

First, how to integrate rich semantic information of a document to obtain entity representations? We construct a document-level graph to integrate complex semantic information, which is a heterogeneous graph containing mention nodes and sentence nodes. Representations of mention nodes and sentence nodes are computed by the pre-trained language model BERT [3]. The built document-level graph is input into the R-GCNs [13], a relational graph neural network, to make nodes contain the information of their neighbor nodes. Then, representations of entities are obtained by performing logsumexp pooling operation on representations of mention nodes. In previous methods, representations of entity nodes are obtained from representations of mention nodes. Hence putting them in the same graph will introduce redundant information and reduce discriminability. Unlike previous document-level graph construction, our document-level graph contains only sentence nodes and mention nodes to avoid redundant information caused by repeated node representations.

Second, how to use connections between entities for reasoning? In this paper, we exploit connections between entities and propose an entity-level graph for reasoning.

The entity-level graph is built by the positional connections between sentences and entities to make full use of cross-sentence information. It connects long-distance cross-sentence entity pairs. Through the learning of GNN, each entity node can aggregate the information of its most relevant entity nodes, which is beneficial to discover potential relations of long-distance cross-sentence entity pairs.

In summary, we propose a novel model called GRACR for document-level RE. Our main contributions are as follows:

- We propose a simplified document-level graph to integrate rich semantic information. The graph contains sentence nodes and mention nodes but not entity nodes, which avoids introducing redundant information caused by repeated node representations.
- We propose an entity-level graph for reasoning to discover potential relations of long-distance cross-sentence entity pairs. An attention mechanism is applied to fuse document embedding, aggregation, and inference information to extract relations of entity pairs.
- We conduct experiments on two public document-level relation extraction datasets. Experimental results demonstrate that our model outperforms many state-of-the-art methods.

2   Related Work

The research on document-level RE has a long history. The document-level graph provides more features for entity pairs. The relevance between entities can be captured through graph learning using GNN [10]. For example, [2] utilized GNN to aggregate the neighborhood information of text graph nodes for text classification. Following this, [1] constructed a document-level graph with heterogeneous nodes and proposed an edge-oriented model to obtain a global representation. [7] characterized the interaction between sentences and entity pairs to improve inter-sentence reasoning. [25] introduced context of entity pairs as edges between entity nodes to model semantic interactions among multiple entities. [24] constructed a dual-tier heterogeneous graph to encode the inherent structure of document and reason multi-hop relations of entities. [17] learned global representations of entities through a document-level graph, and learned local representations based on their contexts. [12] defined the document-level graph as a latent variable to improve the performance of RE models by optimizing the structure of the document-level graph. [23] proposed a double graph-based graph aggregation and inference network (GAIN). Different from GAIN, our entity-level graph is a heterogeneous graph and we use R-GCNs to enable interactions between entity nodes to discover potential relations of long-distance cross-sentence entity pairs. [18] constructed a document-level graph with rhetorical structure theory and used evidence to reasoning. [14] constructed the input documents as heterogeneous graphs and utilized Graph Transformer Networks to generate semantic paths.

Unlike above document-level graph construction methods, our document-level graph contains only sentence nodes and mention nodes to avoid introducing redundant information. Moreover, previous works don't directly deal with cross-sentence entity pairs. Although entities in different sentences are indirectly connected in the graph, e.g., the minimum distance between entities across sentences is 3 and the information needs
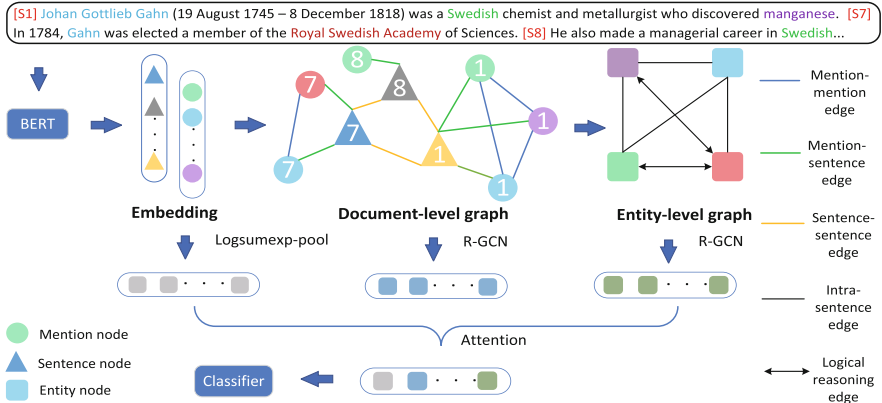
[S1] Johan Gottlieb Gahn (19 August 1745 – 8 December 1818) was a Swedish chemist and metallurgist who discovered manganese. [S7] In 1784, Gahn was elected a member of the Royal Swedish Academy of Sciences. [S8] He also made a managerial career in Swedish...

Fig. 2. Architecture of our proposed model.

to pass through two different nodes when interacting in GLRE [17]. We directly connect cross-sentence entity pairs with potential relations through bridge entities to shorten the distance of information transmission, which reduces the introduction of noise.

In addition, there are some works that try to use pre-trained models directly instead of introducing graph structures. [16] applied a hierarchical inference method to aggregate the inference information of different granularity. [22] captured the coreferential relations in context by a pre-training task. [9] proposed a mention-based reasoning network to capture local and global contextual information. [20] used mention dependencies to construct structured self-attention mechanism. [26] proposed adaptive thresholding and localized context pooling to solve the multi-label and multi-entity problems. These models take advantage of the multi-head attention of Transformer instead of GNN to aggregate information.

However, these studies focused on the local entity representation, which overlooks the interaction between entities distributed in different sentences [11]. To discover potential relations of long-distance cross-sentence entity pairs, we introduce an entity-level graph built by the positional connections between sentences and entities for reasoning.

3   Methodology

In this section, we describe our proposed GRACR model that constructs a document-level graph and an entity-level graph to improve document-level RE. As shown in Fig. 2, GRACR mainly consists of 4 modules: encoding module, document-level graph aggregation module, entity-level graph reasoning module, and classification module. First, in encoding module, we use a pre-trained language model such as BERT [3] to encode the document. Next, in document-level graph aggregation module, we construct a heterogeneous graph containing mention nodes and sentence nodes to integrate rich semantic information of a document. Then, in entity-level graph reasoning module, we also propose a graph for reasoning to discover potential relations of long-distance and cross-sentence entity pairs. Finally, in classification module, we merge the context information of relation representations obtained by self-attention [15] to make final relation prediction.

3.1   Encoding Module

To better capture the semantic information of document, we choose BERT as the encoder. Given an input document $D = [w_1, w_2, \ldots, w_k]$, where $w_j(1 \leq j \leq k)$ is the jth word in it. We then input the document into BERT to obtain the embeddings:

$$\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_k] = \text{Encoder}([w_1, w_2, \ldots, w_k]) \tag{1}$$

where $\mathbf{h}_j \in \mathbb{R}^{d_w}$ is a sequence of hidden states outputted by the last layer of BERT.

To accumulate weak signals from mention tuples, we employ logsumexp pooling [5] to get the embedding e_i^h of entity $\mathbf{e}_i$ as initial entity representation.

$$e_i^h = \log \sum_{j=1}^{N_{\mathbf{e}_i}} \exp \left(h_{\mathbf{m}_j^i} \right) \tag{2}$$

where $\mathbf{m}_j^i$ is the mention $\mathbf{m}_j$ of entity $\mathbf{e}_i$, $h_{\mathbf{m}_j^i}$ is the embedding of $\mathbf{m}_j^i$, $N_{\mathbf{e}_i}$ is the number of mentions of entity $\mathbf{e}_i$ in D.

As shown in Eq. (2), the logsumexp pooling generates an embedding for each entity by accumulating the embeddings of its all mentions across the whole document.

3.2   Document-Level Graph Aggregation Module

To integrate rich semantic information of a document to obtain entity representations, we construct a document-level graph (*Dlg*) based on $\mathbf{H}$.

Dlg has two different kinds of nodes:

Sentence nodes, which represent sentences in D. The representation of a sentence node s_i is obtained by averaging the representations of contained words. We concatenate a node type representation $\mathbf{t}_s \in \mathbb{R}^{d_t}$ to differentiate node types. Therefore, the representations of s_i is $\mathbf{h}_{s_i} = \left[\text{avg}_{w_j \in s_i} (\mathbf{h}_j) ; \mathbf{t}_s \right]$, where $[;]$ is the concatenation operator.

Mention nodes, which represent mentions in D. The representation of a mention node m_i is achieved by averaging the representations of words that make up the mention. We concatenate a node type representation $\mathbf{t}_m \in \mathbb{R}^{d_t}$. Similar to sentence nodes, the representation of m_i is $\mathbf{h}_{m_i} = \left[\text{avg}_{w_j \in m_i} (\mathbf{h}_j) ; \mathbf{t}_m \right]$.

There are three types of edges in *Dlg*:

- Mention-mention edge. To exploit the co-occurrence dependence between mention pairs, we create a mention-mention edge. Mention nodes of two different entities are connected by mention-mention edges if their mentions co-occur in the same sentence.
- Mention-sentence edge. Mention-sentence edge is created to better capture the context information of mention. Mention node and sentence node are connected by mention-sentence edges if the mention appears in the sentence.
- Sentence-sentence edge. All sentence nodes are connected by sentence-sentence edges to eliminate the effect of sentences sequence in the document and facilitate inter-sentence interactions.

Then, we use an L-layer stacked R-GCNs [13] to learn the document-level graph. R-GCNs can better model heterogeneous graph that has various types of edges than GCN. Specifically, its node forward-pass update for the $(l + 1)^{(th)}$ layer is defined as follows:

$$\mathbf{n}_i^{l+1} = \sigma \left(\mathbf{W}_0^l \mathbf{n}_i^l + \sum_{x \in X} \sum_{j \in N_i^x} \frac{1}{|N_i^x|} \mathbf{W}_x^l \mathbf{n}_j^l \right) \tag{3}$$

where $\sigma(\cdot)$ means the activation function, N_i^x denotes the set of neighbors of node i linked with edge x, and X denotes the set of edge types. $\mathbf{W}_x^l, \mathbf{W}_0^l \in \mathbb{R}^{d_n \times d_n}$ are trainable parameter matrices and d_n is the dimension of node representation.

We use the representations of mention nodes after graph convolution to compute the preliminary representation of entity node e_i by logsumexp pooling as e_i^{pre}, which incorporates the semantic information of e_i throughout the whole document. However, the information of the whole document inevitably introduce noise. We employ attention mechanism to fuse the initial embedding information and semantic information of entities to reduce noise. Specifically, we define the entity representation e_i^{Dlg} as follows:

$$e_i^{Dlg} = \mathrm{softmax} \left(\frac{e_i^{pre} \mathbf{W}_i^{e_i^{pre}} \left(e_i^h \mathbf{W}_i^{e_i^h} \right)^T}{\sqrt{d_{e_i^h}}} \right) e_i^h \mathbf{W}_i^{e_i^h} \tag{4}$$

and

$$e_i^{pre} = \log \sum_{j=1}^{N_{\mathbf{e}_i}} \exp \left(n_{m_j^i} \right) \tag{5}$$

where $\mathbf{W}_i^{e_i^{pre}}$ and $\mathbf{W}_i^{e_i^h} \in \mathbb{R}^{d_n \times d_n}$ are trainable parameter matrices. $n_{m_j^i}$ is mention semantic representations after graph convolution. $d_{e_i^h}$ is the dimension of e_i^h.

3.3 Entity-Level Graph Reasoning Module

To discover potential relations of long-distance cross-sentence entity pairs, we introduce an entity-level graph (*Elg*) reasoning module. *Elg* contains only one kind of node:

Entity node, which represents entities in D. The representation of an entity node e_i is obtained from document-level graph defined by Eq. (5). We concatenate a node type representation $\mathbf{t}_e \in \mathbb{R}^{t_e}$. The representations of e_i is $\mathbf{h}_{e_i} = [e_i^{pre}; \mathbf{t}_e]$.

There are two kinds of edges in *Elg*:

- Intra-sentence edge. Two different entities are connected by an intra-sentence edge if their mentions co-occur in the same sentence. For example, *Elg* uses an intra-sentence edge to connect entity nodes e_i and e_j if there is a path $PI_{i,j}$ denoted as $\mathbf{m}_i^{\mathbf{s}_1} \to \mathbf{s}_1 \to \mathbf{m}_j^{\mathbf{s}_1}$. $\mathbf{m}_i^{\mathbf{s}_1}$ and $\mathbf{m}_j^{\mathbf{s}_1}$ are mentions of an entity pair $<\mathbf{e}_i, \mathbf{e}_j>$ and they appear in sentence $\mathbf{s}_1$. "$\to$" denotes one reasoning step on the reasoning path from entity node e_i to e_j.
- Logical reasoning edge. If the mention of entity $\mathbf{e}_k$ has co-occurrence dependencies with mentions of other two entities in different sentences, we suppose that $\mathbf{e}_k$ can be used as a bridge between entities. Two entities distributed in different sentences are connected by a logical reasoning edge if a bridge entity connects them. There is a

logical reasoning path $PL_{i,j}$ denoted as $\mathbf{m}_i^{\mathbf{s}_1} \rightarrow \mathbf{s}_1 \rightarrow \mathbf{m}_k^{\mathbf{s}_1} \rightarrow \mathbf{m}_k^{s_2} \rightarrow \mathbf{s}_2 \rightarrow \mathbf{m}_j^{\mathbf{s}_2}$, and we apply a logical reasoning edge to connect entity nodes e_i and e_j.

Similar to *Dlg*, we apply an L-layer stacked R-GCNs to convolute the entity-level graph to get the reasoned representation of entity e_i^{Elg}. In order to better integrate the information of entities, we employ the attention mechanism to fuse the aggregated information, the reasoned information, and the initial information of entity to form the final representation of entity.

$$e_i^{rep} = \mathrm{softmax}\left(\frac{e_i^{Dlg}\mathbf{W}_i^{e_i^{Dlg}}\left(e_i^{Elg}\mathbf{W}_i^{e_i^{Elg}}\right)^T}{\sqrt{d_{e_i^{Elg}}}}\right)e_i^h\mathbf{W}_i^{e_i^h} \tag{6}$$

where $\mathbf{W}_i^{e_i^{Dlg}}$ and $\mathbf{W}_i^{e_i^{Elg}} \in \mathbb{R}^{d_n \times d_n}$ are trainable parameter matrices. $d_{e_i^{Elg}}$ is the dimension of e_i^{Elg}.

3.4 Classification Module

To classify the target relation r for an entity pair $<e_m, e_n>$, we concatenate entity final representations and relative distance representations to represent one entity pair:

$$\hat{e}_m = [e_m^{rep}; s_{mn}], \hat{e}_n = [e_n^{rep}; s_{nm}] \tag{7}$$

where s_{mn} denotes the embedding of relative distance from the first mention of e_m to that of e_n in the document. s_{nm} is similarly defined.

Then, we concatenate the representations of $\hat{e}_m$, $\hat{e}_n$ to form the target relation representation $\mathbf{o}_r = [\hat{e}_m; \hat{e}_n]$.

Furthermore, following [17], we employ self-attention [15] to capture context relation representations, which can help us exploit the topic information of the document:

$$\mathbf{o}_c = \sum_{i=1}^{p} \theta_i \mathbf{o}_i = \sum_{i=1}^{p} \frac{\exp\left(\mathbf{o}_i \mathbf{W}\mathbf{o}_r^T\right)}{\sum_{j=1}^{p}\exp\left(\mathbf{o}_j\mathbf{W}\mathbf{o}_r^T\right)}\mathbf{o}_i \tag{8}$$

where $\mathbf{W} \in \mathbb{R}^{d_r \times d_r}$ is a trainable parameter matrix, d_r is the dimension of target relation representations. $\mathbf{o}_i$ is the relation representation of the ith entity pair. θ_i is the attention weight for $\mathbf{o}_i$. p is the number of entity pairs.

Finally, we use a feed-forward neural network (FFNN) on the target relation representation o_r and the context relation representation o_c for prediction. What's more, we transform the multi-classification problem into multiple binary classification problems, since an entity pair may have different relations. The predicted probability distribution of r over the set R of all relations is defined as follows:

$$y_r = \mathrm{sigmoid}\left(\mathrm{FFNN}\left([\mathbf{o}_r; \mathbf{o}_c]\right)\right) \tag{9}$$

where $y_r \in \{0, 1\}$.

We define the loss function as follows:

$$L = -\sum_{r \in R}\left(y_r^* \log\left(y_r\right) + \left(1 - y_r^*\right)\log\left(1 - y_r\right)\right) \tag{10}$$

where $y_r^* \in \{0, 1\}$ denotes the true label of r.

4   Experiments and Results

Table 1. Statistics of the datasets.

Statistics	DocRED	CDR
# Train	3053	500
# Dev	1000	500
# Test	1000	500
# Relations	97	2
Avg.# Ents per Doc.	19.5	7.6

Table 2. Results on the development and test set of DocRED. Some results are quoted from respective paper.

Model		Dev		Test	
		Ign F_1	F_1	Ign F_1	F_1
Sequence-based	CNN [21]	41.58	43.45	40.33	42.26
	LSTM [21]	48.44	50.68	47.71	50.07
	BiLSTM [21]	48.87	50.94	48.78	51.06
	Context-aware [21]	48.94	51.09	48.40	50.70
Transformer-based	BERT [19]	–	54.16	–	53.20
	HIN [16]	54.29	56.31	53.70	55.60
	CorefBERT [22]	55.32	57.51	54.54	56.96
	SSAN [20]	57.03	59.19	55.84	58.16
Graph-based	EoG [1]	45.94	52.15	49.48	51.82
	GEDA [7]	54.52	56.16	53.71	55.74
	GCGCN [25]	55.43	57.35	54.53	56.67
	GLRE [17]	–	–	55.40	57.40
	DISCO [18]	55.91	57.78	55.01	55.70
Ours	GRACR	**57.85**	**59.73**	**56.47**	**58.54**

Table 3. Results on CDR.

Model	F_1	intra-F_1	inter-F_1
LSR [12]	64.8	68.9	53.1
DHG [24]	65.9	70.1	54.6
HGNN [14]	64.4	69.2	51.2
MRN [9]	65.9	70.4	54.2
GRACR	**68.8**	**73.9**	**55.8**

Table 4. Ablation study on the development set of DocRED.

Model	Ign F_1	F_1
GRACR	**57.85**	**59.73**
w/o both module	57.33	59.16
w/o reasoning module	57.44	59.30
w/o aggregation module	57.61	59.57
w/o reasoning edge	57.52	59.48
w/o intra-sentence edge	57.51	59.46
w/ previous Dlg	57.13	58.97

Table 5. Intra-F_1 and inter-F_1 results on DocRED.

Model	intra-F_1	inter-F_1
CNN [21]	51.87	37.58
LSTM [21]	56.57	41.47
BiLSTM [21]	57.05	43.39
Context-aware [21]	56.74	42.26
GEDA [7]	61.85	49.46
LSR [12]	65.26	52.05
GRACR	**65.88**	**52.49**

4.1 Dataset

We evaluate our model on DocRED and CDR dataset. The dataset statistics are shown in Table 1. The DocRED dataset [21], a large-scale human-annotated dataset constructed from Wikipedia, has 3,053 documents, 132,275 entities, and 56,354 relation facts in total. DocRED covers a wide variety of relations related to science, art, time, personal life, etc. The Chemical-Disease Relations (CDR) dataset [8] is a human-annotated dataset, which is built for the BioCreative V challenge. CDR contains 1,500 PubMed abstracts about chemical and disease with 3,116 relational facts.

4.2 Experiment Settings and Evaluation Metrics

To implement our model, we choose uncased BERT-base [3] as the encoder on DocRED and set the embedding dimension to 768. For CDR dataset, we pick up BioBERT-Base v1.1 [6], which re-trained the BERT-base-cased model on biomedical corpora.

All hyper-parameters are tuned based on the development set. Other parameters in the network are all obtained by random orthogonal initialization [17] and updated during training.

For a fair comparison, we follow the same experimental settings from previous works. We apply F_1 and Ign F_1 as the evaluation metrics on DocRED. F_1 scores can be obtained by calculation through an online interface. Furthermore, Ign F_1 means that the F_1 score ignores the relational facts shared by the training and development/test sets. We compare our model with three categories of models. Sequence-based models use neural architectures such as CNN and bidirectional LSTM as encoder to acquire embeddings of entities. Graph-based models construct document graphs and use GNN to learn graph structures and implement inference. Instead of using document graph, transformer-based models adopt pre-trained language models to extract relation.

For CDR dataset, we use training subset to train the model. Depending on whether relation between two entities occur within one sentence or not, F_1 can be further split into intra-F_1 and inter-F_1 to evaluate the model's performance on intra-sentence relations and inter-sentence relations. To make a comprehensive comparison, we also measure the corresponding F_1, intra-F_1 and inter-F_1 scores on development set.

4.3   Main Results

Results on DocRED. As shown in Table 2, our model outperforms all baseline methods on both development and test sets. Compared with graph-based models, both F_1 and Ign F_1 of our model are significantly improved. Compared to GLRE, which is the most relevant approach to our method, the performance improves 1.07% for F_1 and 1.14% for Ign F_1 on test set. Furthermore, compared to Transformer-based model SSAN, our method improves by 0.54% for F_1 and 0.84% for Ign F_1 on development set. With respect to sequence-based methods, the improvement is considerable.

Results on CDR. Table 3 depicts the comparisons with state-of-the-art models on CDR. Compared to MRN [9], the performance of our model approximately improves about 2.9% for F_1, and 3.9% for intra-F_1 and 1.6% for inter-F_1. DHG and MRN produce similar results. In summary, these results demonstrate that our method is effective in extracting both intra-sentence relations and inter-sentence relations.

Johan Gottlieb Gahn
[S1] Johan Gottlieb Gahn (19 August 1745 − 8 December 1818) was a Swedish chemist ···[S7] In 1784, Gahn was elected a member of the Royal Swedish Academy of Sciences. [S8] He also made a managerial career in Swedish mining.
Ground Truth & Evidence:
(1) <Johan Gottlieb Gahn, country of citizenship, Swedish> , [S1, S7, S8]
(2) <Johan Gottlieb Gahn, member of, Royal Swedish Academy>, [S7]
(3) <Royal Swedish Academy, country, Swedish>, [S1, S7]
Result   Ours: (1), (2), (3)   GLRE: (1), (2)

Fig. 3. Case study on the DocRED development set. Entities are colored accordingly.

4.4   Ablation Study

We conduct a thorough ablation study to investigate the effectiveness of two key modules in our method: an aggregation module and an reasoning module. From Table 4, we can observe that all components contribute to model performance.

(1) When the reasoning module is removed, the performance of our model on the DocRED development set for Ign F_1 and F_1 scores drops by 0.41% and 0.43%, respectively. Furthermore, we analyze the role of each edge in the reasoning module. F_1 drops by 0.23% or 0.25% when we remove intra-sentence edge or logical reasoning edge. Likewise, removing the aggregation module results in 0.24% and 0.16% drops in Ign F_1 and F_1. This phenomenon verifies the effectiveness of the aggregation module and the reasoning module.

(2) A larger drop occurs when two modules are removed. The F_1 score dropped from 59.73% to 59.16% and the Ign F_1 score dropped from 57.85% to 57.33%. This study validates that all modules work together can handle RE task more effective.

(3) When we apply the document-level graph with entity nodes and more complex edge types like GLRE, the F_1 score dropped from 59.73% to 58.97% and the Ign F_1 score dropped from 57.85% to 57.13%. This result suggests that document-level graph containing complex and repetitive node information and edges can lead to information redundancy and degrade model performance.

4.5   Intra- and Inter-sentence Relation Extraction

In this subsection, we further analyze both intra- and inter-sentence RE performance on DocRED. The experimental results are listed in Table 5, from which we can find that GRACR outperforms the compared models in terms of intra- and inter-F_1. For example, our model obtains 0.62% intra-F_1 and 0.44% inter-F_1 gain on DocRED. The improvements suggest that GRACR not only considers intra-sentence relations, but also handles long-distance inter-sentence relations well.

4.6   Case Study

As shown in Fig. 3, GRACR infers the relations of <Swedish, Royal Swedish Academy of Sciences> based on the information of $S1$ and $S7$. "Swedish" and "Royal Swedish Academy of Sciences" distributed in different sentences are connected by entity-level graph because they appear in the same sentence with "Johan Gottlieb Gahn". Entity-level graph connects them together to facilitate reasoning about their relations. More importantly, our method is in line with the thinking of human logical reasoning. For example, from ground true we can know that "Gahn"'s country is "Swedish". Therefore, we can speculate that there is a high possibility that the organization he joined has a relation with "Swedish".

5   Conclusion

In this paper, we propose GRACR, a graph information aggregation and logical cross-sentence reasoning network, to better cope with document-level RE. GRACR applies a document-level graph and attention mechanism to model the semantic information of all mentions and sentences in a document. It also constructs an entity-level graph to utilize the interaction among different entities to reason the relations. Finally, it uses an attention mechanism to fuse document embedding, aggregation, and inference information to help identify relations. Experimental results show that our model achieves excellent performance on DocRED and CDR.

References

1. Christopoulou, F., Miwa, M., Ananiadou, S.: Connecting the dots: document-level neural relation extraction with edge-oriented graphs. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 4925–4936 (2019)
2. Dai, Y., Shou, L., Gong, M., Xia, X., Kang, Z., Xu, Z., Jiang, D.: Graph fusion network for text classification. Knowl.-Based Syst. **236**, 107659 (2022)

3. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics, pp. 4171–4186 (2019)

4. Fang, R., Wen, L., Kang, Z., Liu, J.: Structure-preserving graph representation learning. In: ICDM (2022)

5. Jia, R., Wong, C., Poon, H.: Document-level n-ary relation extraction with multiscale representation learning. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 3693–3704 (2019)

6. Lee, J., et al.: Biobert: a pre-trained biomedical language representation model for biomedical text mining. Bioinformatics **36**(4), 1234–1240 (2020)

7. Li, B., Ye, W., Sheng, Z., Xie, R., Xi, X., Zhang, S.: Graph enhanced dual attention network for document-level relation extraction. In: Proceedings of the 28th International Conference on Computational Linguistics, pp. 1551–1560 (2020)

8. Li, J., Sun, Y., Johnson, R.J., Sciaky, D., Wei, C.H., Leaman, R., Davis, A.P., Mattingly, C.J., Wiegers, T.C., Lu, Z.: Biocreative v cdr task corpus: a resource for chemical disease relation extraction. Database **1**, 10 (2016)

9. Li, J., Xu, K., Li, F., Fei, H., Ren, Y., Ji, D.: 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, pp. 1359–1370 (2021)

10. Liu, L., Kang, Z., Ruan, J., He, X.: Multilayer graph contrastive clustering network. Inf. Sci. **613**, 256–267 (2022)

11. Luoma, J., Pyysalo, S.: Exploring cross-sentence contexts for named entity recognition with Bert. In: Proceedings of the 28th International Conference on Computational Linguistics, pp. 904–914 (2020)

12. Nan, G., Guo, Z., Sekulić, I., Lu, W.: Reasoning with latent structure refinement for document-level relation extraction. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 1546–1557 (2020)

13. Schlichtkrull, M., Kipf, T.N., Bloem, P., van den Berg, R., Titov, I., Welling, M.: Modeling relational data with graph convolutional networks. In: Gangemi, A., et al. (eds.) ESWC 2018. LNCS, vol. 10843, pp. 593–607. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93417-4_38

14. Shi, Y., Xiao, Y., Quan, P., Lei, M., Niu, L.: Document-level relation extraction via graph transformer networks and temporal convolutional networks. Pattern Recogn. Lett. **149**, 150–156 (2021)

15. Sorokin, D., Gurevych, I.: Context-aware representations for knowledge base relation extraction. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 1784–1789 (2017)

16. Tang, H., et al.: HIN: hierarchical inference network for document-level relation extraction. In: Lauw, H.W., Wong, R.C.-W., Ntoulas, A., Lim, E.-P., Ng, S.-K., Pan, S.J. (eds.) PAKDD 2020. LNCS (LNAI), vol. 12084, pp. 197–209. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-47426-3_16

17. Wang, D., Hu, W., Cao, E., Sun, W.: Global-to-local neural networks for document-level relation extraction. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 3711–3721 (2020)

18. Wang, H., Qin, K., Lu, G., Yin, J., Zakari, R.Y., Owusu, J.W.: Document-level relation extraction using evidence reasoning on RST-graph. Knowl.-Based Syst. **228**, 107274 (2021)

19. Wang, H., Focke, C., Sylvester, R., Mishra, N., Wang, W.: Fine-tune Bert for docred with two-step process. arXiv preprint arXiv:1909.11898 (2019)

20. Xu, B., Wang, Q., Lyu, Y., Zhu, Y., Mao, Z.: Entity structure within and throughout: Modeling mention dependencies for document-level relation extraction. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 14149–14157 (2021)

21. Yao, Y., et al.: Docred: A large-scale document-level relation extraction dataset. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 764–777 (2019)

22. Ye, D., et al.: Coreferential reasoning learning for language representation. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 7170–7186 (2020)

23. Zeng, S., Xu, R., Chang, B., Li, L.: Double graph based reasoning for document-level relation extraction. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1630–1640 (2020)

24. Zhang, Z., et al.: Document-level relation extraction with dual-tier heterogeneous graph. In: Proceedings of the 28th International Conference on Computational Linguistics, pp. 1630–1641 (2020)

25. Zhou, H., Xu, Y., Yao, W., Liu, Z., Lang, C., Jiang, H.: Global context-enhanced graph convolutional networks for document-level relation extraction. In: Proceedings of the 28th International Conference on Computational Linguistics, pp. 5259–5270 (2020)

26. Zhou, W., Huang, K., Ma, T., Huang, J.: Document-level relation extraction with adaptive thresholding and localized context pooling. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 14612–14620 (2021)