Applied Intelligence (2024) 54:5353–5372
https://doi.org/10.1007/s10489-024-05448-4

Graph neural networks with selective attention and path reasoning for document-level relation extraction

Tingting Hang[1] · **Jun Feng**[2,3] · **Yunfeng Wang**[2,3] · **Le Yan**[4]

Accepted: 7 April 2024 / Published online: 20 April 2024

Abstract
Document-level Relation Extraction (DocRE) aims to extract relations from multiple sentences simultaneously. Existing graph-based methods adopt static graphs to represent the document structure, which is unable to capture complex interactions. Besides, they take all sentences in the document as the scope of relation extraction (RE) while introducing noise by irrelevant sentences. Furthermore, they do not explicitly model the reasoning chain, leading to a lack of explainability in the reasoning results. These limitations may significantly hinder their performance in practical applications. In this paper, we propose a model based on selective attention and path reasoning for DocRE. Firstly, we adopt hierarchical heterogeneous graph neural networks and recurrent neural networks to realize document modeling and capture complex interactions in the document. Secondly, we adopt selective attention to select sentences related to the entity pair to generate document subgraphs as the scope of RE. Lastly, we adopt path reasoning to explicitly model the reasoning chain between multiple entities in the document subgraph, infer the relations between entities and provide corresponding supporting evidence. Extensive experiment results on three benchmark datasets show that the proposed framework is effective and achieves superior performance compared to most methods. Further analysis demonstrates that selective attention and path reasoning can discover more accurate inter-sentence relations and supporting evidence.

Keywords Graph neural network · Selective attention · Path reasoning · Document-level relation extraction · Supporting evidence

1 Introduction

Relation extraction (RE) aims to identify semantic relations between entities in text. It plays an important role in many natural language processing applications, such as knowledge graph construction [1] and automatic question answering [2]. Previous studies mainly focused on sentence-level RE [3–9], which requires a sentence to contain two entities. However, sentence-level RE models suffer from an inevitable limitation – they fail to recognize relations between entities across sentences. Hence, recent studies have been moving towards the more realistic setting of document-level relation extraction (DocRE).

DocRE requires reading and reasoning over multiple sentences in a document, aiming to extract all possible relation instances from a document and provide supporting evidence for them. As shown in Fig. 1, the task is to extract the relation between "Riddarhuset" and "Sweden". We must first identify that "Riddarhuset" is located in "Stockholm" from sentence 4, and then identify that "Stockholm" is the capital of "Sweden" from sentence 1. From the reasoning chain "located in – capital" of the entity pair (Riddarhuset, Sweden), it can also be predicted that the sovereign state of "Riddarhuset" is "Sweden". Sentences 1 and 4 provide supporting evidence for this relation. Other sentences do not mention two entities,

✉ Tingting Hang
hangtt@ahut.edu.cn

Jun Feng
fengjun@hhu.edu.cn

Yunfeng Wang
naive@hhu.edu.cn

Le Yan
yanle@hhu.edu.cn

[1] School of Computer Science and Technology, Anhui University of Technology, Ma'anshan, China

[2] Key Laboratory of Water Big Data Technology of Ministry of Water Resource, Hohai University, Nanjing, China

[3] School of Computer and Information, Hohai University, Nanjing, China

[4] College of Information Engineering, Nanjing Xiaozhuang University, Nanjing, China

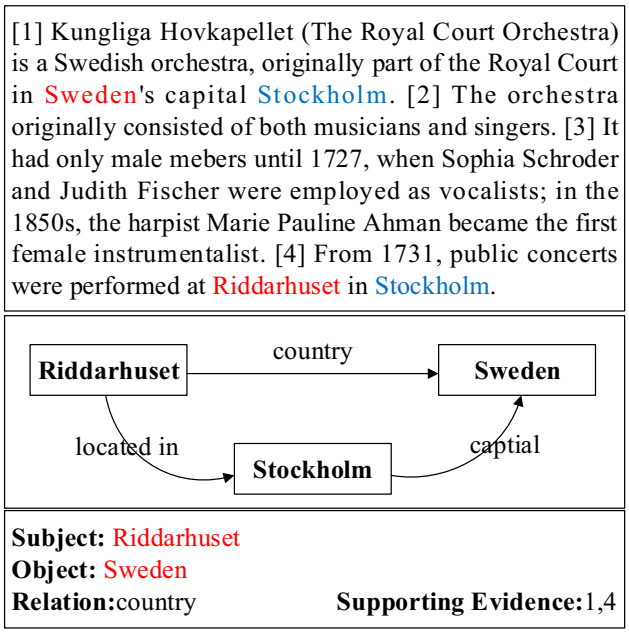


Fig. 1 An example excerpted from the DocRED dataset, in which the head and tail entity pair spans multiple sentences. Red represents the head and tail entities, and blue represents the bridge entity required for logical reasoning

it is irrelevant to the relation prediction. Sometimes including such irrelevant sentences in the input might introduce noise to the model.

Among the techniques available in the literature, a simple way is the sequence-based methods. Concretely, entity embeddings are obtained by encoding the entire document and involving a bilinear classifier to predict the relation. Such as CNN[10], LSTM[11], BiLSTM[12], Context-aware[13] etc. Another popular strand of this field uses mentions, entities, and sentences as nodes to construct the document graph, and explicitly learn the associations between entities through graph propagation, thereby realizing the relational reasoning between entities [14–19]. Recently, some works [20–24] relies on the transformer architecture to model cross-sentence relations since transformers can implicitly capture long-distance dependencies. Such models do not need to introduce the document graph, and can automatically learn the edges of dependency structures and coreference structures. However, there are three limitations of the existing DocRE methods. Firstly, existing graph-based methods utilize static graphs to represent the structure of the entire document, which is unable to capture complex interactions in the document. Secondly, existing methods utilize the entire document sentences for RE, and the noise problem brought by irrelevant sentences is not addressed. Lastly, there are very few works that explicitly model the reasoning chain, resulting in a lack of explainability in the extraction results.

Based on the above observations, we propose a DocRE model to overcome the challenges, which automatically extracts relation and evidence in three stages. In the first stage, we adopt hierarchical heterogeneous graph neural networks to construct the document graph for document representation and then apply recurrent neural networks (RNN) to capture local and non-local interactions in the document. In the second stage, we adopt selective attention to select sentences related to the entity pair from the document graph and aggregate all related sentences to generate document subgraphs. In the last stage, we adopt a path reasoning mechanism to explicitly model the reasoning chain between multiple entities in the document subgraph, thus inferring the multi-hop relation and providing supporting evidence.

We conduct extensive experiments on three public widely used DocRE datasets. Experiment results show that our model outperforms most baseline models. Besides, we provide a detailed analysis demonstrating that selective attention and path reasoning can discover more accurate inter-sentence relations and supporting evidence. The advantages of our method are summarized as follows:

- We combine hierarchical heterogeneous graph neural networks and recurrent neural networks into the document learning framework, which allows the models to construct the document graph and capture complex interactions in the document.
- We propose a selective attention subgraph module for select sentences related to the entity pair, which reduces the interference of irrelevant sentences.
- We propose a path reasoning module to model the reasoning chain between multiple entities in the subgraph, which increases the explainability of the extraction results.

The rest of the paper is organized as follows. In Section 2, we review related works. The proposed model is described in Section 3. In Section 4, we introduce the relevant content of the experiments. In Section 5, we analyze and discuss the experiment results from different perspectives. The conclusions of the paper are drawn in Section 6.

2 Related work

Early studies [25–27] confined DocRE to short text spans (e.g., three consecutive sentences) while ignoring the relational reasoning in the document. Recent work has expanded this range to the entire document of the biomedical domain [14, 15, 24, 28]. Zhang et al. [28] present a novel graph-based approach for DocRE with a Dual-tier Heterogeneous Graph (DHG), to achieve document modeling and multi-hop reasoning in proper order. Xiao et al. [24] proposed to explic-

itly teach the model to capture textual contexts and entity types by Supervising and Augmenting Intermediate Steps (SAIS). However, the datasets used in these methods contain very limited relations and entity types. Owing to the introduction of large-scale document-level datasets, some researchers began to study large-scale DocRE. These DocRE methods can be grouped into Sequence-based, Graph-based, and Transformer-based methods.

Sequence-based methods Sequence-based methods obtain entity embeddings by encoding the entire document and then adopt a bilinear classifier to predict the relation between entity pairs [10–13]. However, these methods do not consider the coreference relation of multiple mentions in the document and do not perform any reasoning, which affects the extraction performance of inter-sentence relations.

Graph-based methods Graph-based methods use mentions, entities, and sentences as nodes to construct the document graph, and explicitly learn the associations between entities through graph propagation, thereby realizing the relational reasoning between entities [14–19]. Wang et al. [16] proposed a Gobal-to-Local neural network for document-level RE (GLRE) that encodes the document information in terms of entity global and local representations as well as context relation representations. GLRE is particularly effective in extracting relations between entities of long distance and having multiple mentions. Nan et al. [18] proposed Latent Structure Refinement (LSR) generates latent document graphs, while further developing an optimization strategy that enables the model to gradually aggregate relevant information for multi-hop reasoning. LSR placed the mention node and the entity node within the same graph and conducted reasoning implicitly using a GCN, which can discover more accurate inter-sentence relations. However, the results inferred by LSR lacked explainability. Zeng et al. [19] proposed Graph Aggregation and Inference Network (GAIN). GAIN first constructs both a mention-level graph and an entity-level graph and then perform multi-hop reasoning on both graphs. However, the complicated operations on the graphs lower the efficiency of these methods.

Transformer-based methods Different from the above methods, Transformer-based methods do not need to introduce the document graph, and can automatically learn edges of dependency structures and coreference structures by Pre-trained Language Models (PLM)[20–24]. Tang et al. [20] proposed a Hierarchical Inference Network (HIN) that fully exploits the abundant information from the entity, sentence, and document levels to perform relational reasoning. However, HIN is unable to capture the structural dependencies between entities. Zhou et al. [21] proposed Adaptive Thresholding and Localized Context Pooling (ATLOP) that solve multi-label and multi-entity problems. However, ATLOP neglected the interdependencies among the multiple relations. Xu et al. [22] proposed a Structured Self-attention Network (SSAN) that incorporates the structural dependencies within the standard self-attention mechanism and the overall encoding stage. SSAN performs contextual reasoning and structural reasoning simultaneously and interactively, which substantially improves the performance of RE tasks. Zhang et al. [23] views DocRE as a semantic segmentation task. However, these studies focus on the extraction of relational facts in the document and have not extended to supporting evidence extraction.

In this paper, we propose a novel DocRE model that utilizes the advantages of selective attention and path reasoning to guide the RE. Compared to other graph-based methods, our architecture features many different designs. Firstly, the ways of document graph construction are different. We adopt hierarchical heterogeneous graph neural networks to construct the document graph, then adopts RNN to capture complex interactions in the document. While other methods adopt static graphs to represent the document structure, which is unable to capture complex interactions. Secondly, computational complexity is different. Other methods (e.g., GAIN) take all sentences in the document as the scope of RE. Instead, we adopt selective attention to select the key sentence for RE while reducing the noise impact of irrelevant sentences. Finally, the process of path reasoning is different. Other methods do not explicitly model the reasoning chain. Instead, we adopt path reasoning to model the reasoning chain explicitly without extra overhead, and the sentences appearing on the reasoning path serve as supporting evidence, thus providing explainability for the extraction results.

3 The proposed method

3.1 Model formulation

We formulate the DocRE task as follows. Given an annotated document $\mathcal{D}=\{s_l\}_{l=1}^{L}$, its entity set $V^e=\{e_i\}_{i=1}^{n_e}$, and the set of relations $\mathcal{R}$, where $s_l=\{x_k\}_{k=1}^{C}$ denotes the l-th sentence with C words and $e_i=\{m_k\}_{k=1}^{n_{e_i}}$ is the i-th entity with n_{e_i} entity mentions. The DocRE task does not require giving a head entity and a tail entity, the model needs to predict all intra- and inter-sentence relations between different entities in V^e, namely $\{(e_i, r_{ij}, e_j)|e_i, e_j \in V^e, r_{ij} \in R, i \neq j\}$, along with the evidence sentences $E_{r_{i,j}}=\{s_k\}_{k=1}^{m}$ that are supporting these relation instances.

3.2 Model architecture

The framework of SAPR-GNN contains three modules: *GNN–RNN module* (Section 3.2.1), *Selective attention subgraph module* (Section 3.2.2) and *Path reasoning module*

(Section 3.2.3). An illustration of our framework is shown in Fig. 2.

The main workflow comprises three steps:

Step 1: The GNN–RNN module hierarchically constructs intra-sentence and inter-sentence relation graphs. It utilizes GNN as the encode to enhance the vector representation of mentions and sentences. On this basis, it uses RNN to capture the local and non-local information interaction in the document.

Step 2: The selective attention subgraph module utilizes selective attention to select sentences related to the entity pair and generate document subgraphs as the scope of RE.

Step 3: The path reasoning module explicitly models the reasoning chain in the document subgraph, and predicts the probability of each relation in a given relation path, thereby realizing the extraction of relation instances and supporting evidence.

As shown in Fig. 2, the document graph captures various types of dependencies through different types of nodes and edges. The node is composed of three parts: mention, entity, and sentence. The five types of edges are shown in Table 1.

3.2.1 GNN–RNN module

The GNN–RNN module constructs a two-layer heterogeneous graph $G = (\{G_{1(l)}\}_{l=1}^{L}, G_2)$ to represent the intra-sentence and inter-sentence information in the document. The input of this module is the document $\mathcal{D}$, the entity V^e, and its corresponding mention V^x, and the output is the document graph G and the final representation H_l of each sentence. The main workflow contains three steps:

Step 1: The intra-sentence relation graph G_1 is comprised of entity mentions of each sentence, and it outputs the mention vector representation h_i.

Step 2: The inter-sentence relation graph G_2 is comprised of sentences in the document, and it outputs the sentence vector representation g_l.

Step 3: Aggregating G_1 and G_2 to form a document graph G, and outputing the final representation H_l of each sentence, whose value is comprised of the mention vector representation and the sentence vector representation. Through the recurrent state transition process of RNN, H_l can capture the information for the entire document.

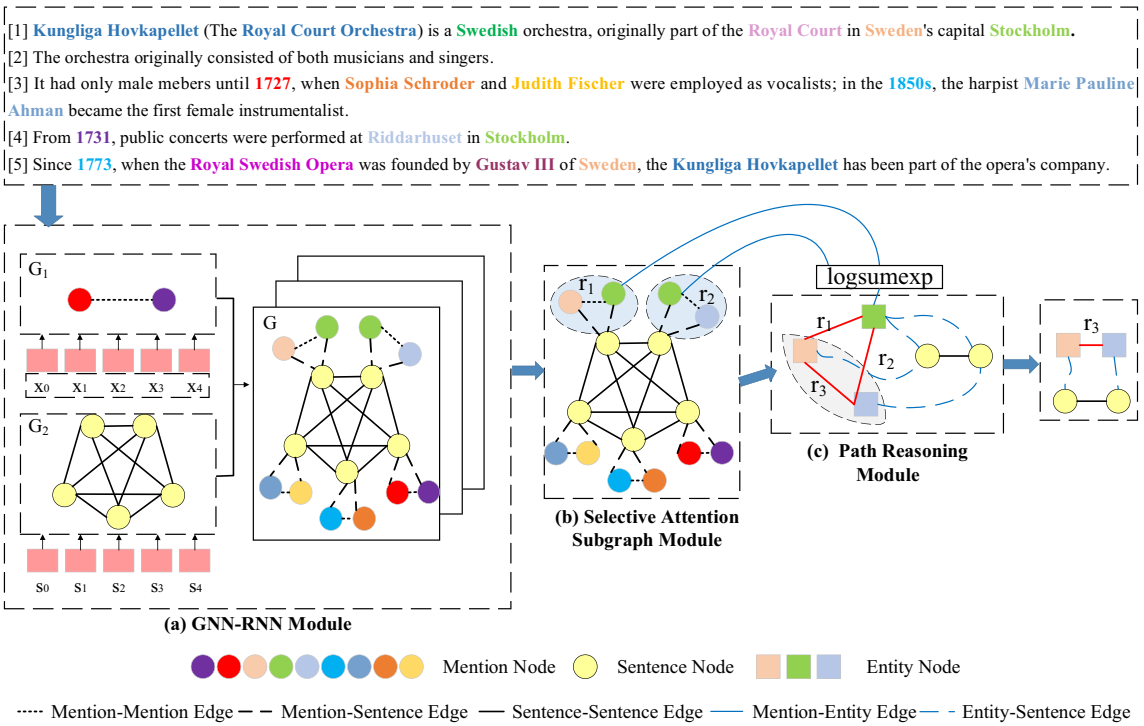


Fig. 2 The Overview of our proposed method. The model takes the entire document as input, and first applies the GNN–RNN module to generate a document graph, and then applies the selective attention module to generate document subgraphs. Finally, applies the path reasoning module to identify semantic relations and supporting evidence in the document subgraph. The yellow nodes represent sentences, and the nodes in other colors represent entities and their corresponding mentions

Table 1 Different types of edges in the document graph

Edge	Describtion
Mention-Mention Edge	An edge is established between two mention nodes if they appear together in a sentence.
Mention-Sentence Edge	An edge is established between a mention node and a sentence node if the mention node appears in the sentence.
Mention-Entity Edge	An edge is established between a mention node and an entity node If the mention refers to the entity.
Entity-Sentence Edge	An edge is established between an entity node and a sentence node If the mention of the entity appears in the sentence.
Sentence-Sentence Edge	Build connections for any two sentence nodes to model sequential and non-sequential information.

An example of the document graph construct process is shown in Fig. 3. Firstly, three intra-sentence relation graph is constructed with mention nodes of each sentence. Then, an inter-sentence relation graph is constructed with sentence nodes in the same document. Finally, the above graphs are aggregated together to generate the document graph.

(1) Intra-sentence relation graph

We construct a fully connected intra-sentence relation graph $G_1 = (V^x, E^x)$, where V^x denotes the set of mentions in the sentence, and each edge $(h_i, h_j) \in E^x$ denotes the relation between the mention pair. In this section, we will introduce how to construct G_1.

Context-enhance word representation The context-enhance word representation embeds both semantic and augmented information of words into their word representations. To be more specific, we use wording embedding as a basic feature to capture meaningful semantic regularities. Meanwhile, coreference, entity type, sentence number, and word position embeddings are also used to augment the representation. We directly use *nn.embedding* to initialize the embedding matrix of augmented information. Specifically, *torch.nn.init.xavier_uniform_* is used to fill each position in the embedding matrix with a *xavier_uniform* initialization value, and these values are subject to a uniform distribution.

$$E(x_k^{i,j}) = [w_k; c_k; t_k; n_k; u_k^{i,j}], \tag{1}$$

where w_k, c_k, t_k, and n_k denotes the word, coreference, entity type, and sentence number embeddings of word x_k, respectively; and $u_k^{i,j}$ denotes the position embedding of word x_k relative to the mention pair (h_i, h_j). These embeddings are described below.

- *Word embedding*: Each word is mapped to a d_w-dimensional vector by a word embedding matrix $P \in \mathbb{R}^{|V_w| * d_w}$, where $|V_w|$ is the size of the vocabulary and d_w is the dimension of word embedding. For DocRED, CDR, and GDA, we used GloVe pre-trained word embeddings [29], PubMed pre-trained word embeddings [30], and randomly initialized word embeddings [15], respectively.

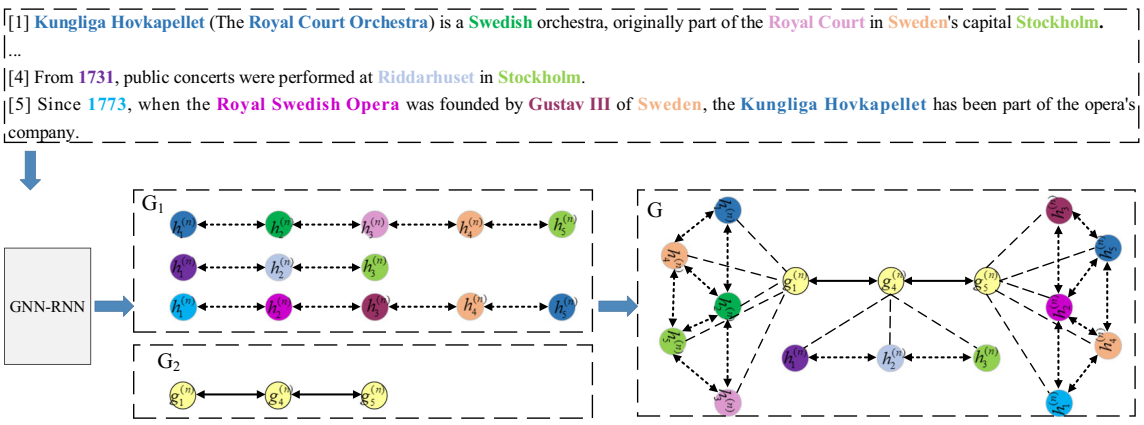


Fig. 3 An example of the document graph construction process. For a good visualization, we only exhibit some representative edges and nodes in this figure

- *Coreference embedding*: Mentions corresponding to the same entity are assigned the same entity ID, which is determined by the order in which the entity appears in the document. Each entity ID is mapped to a d_c-dimensional vector by a coreference embedding matrix $P \in \mathbb{R}^{|V_c|*d_c}$, where $|V_c|$ is the number of entities, and d_c is the dimension of coreference embedding.
- *Entity type embedding.*: Each label type is mapped to a d_t-dimensional vector by an entity-type embedding matrix $P \in \mathbb{R}^{|V_t|*d_t}$, where $|V_t|$ is the number of entity types, and d_t is the dimension of entity-type embedding.
- *Sentence number embedding.*: To facilitate the integration of G_1 and G_2, a sentence number embedding is added to each word to indicate which sentence the word belongs to.
- *Word position embedding*: Following the method proposed in [13], each word in the sentence is marked as either belonging to the first or second mentions or neither. Each position marker is mapped to a d_{p_w}-dimensional vector by a position embedding matrix $P \in \mathbb{R}^{3*d_{pw}}$, where d_{p_w} is the dimension of word position embedding.

Mention pair edge representation Following the method proposed in [6], we feed the representations of mention pairs into the encoder, which converts sequences into transition matrices corresponding to edges. The encoder contains a bidirectional long short-term memory (BiLSTM) and a multi-layer perceptron (MLP) or BERT. Taking the former as an example,

$$\varsigma_{i,j} = MLP\left(BiLSTM\left(s_{i,j}\right)\right), \tag{2}$$

where $s_{i,j} = \{E(x_k^{i,j})\}_{k=1}^{C}$ denotes the representation of the mention pair, C denotes the number of words in the sentence, and $\varsigma_{i,j}$ is the edge representation of the mention pair.

Multi-layer learned mention representation The edge representation of mention pairs is the input to the multi-layer RNN, and the mention vector representation is learned layer by layer:

$$h_i^{(n)} = \sigma(\sum\nolimits_{j\in N(i)} \varsigma_{i,j}^{(n-1)} h_j^{(n-1)} + W_h{}^{(n-1)} h_i^{(n-1)}), \tag{3}$$

where $h_i^{(n)}$ denotes the hidden vector of the mention node i in the sentence at n-layer, σ denotes the nonlinear activation function, $N(i)$ denotes the set of neighbor nodes of the mention node i, and $W_h^{(n-1)}$ denotes a trainable parameter. The superscript (n) is the RNN layer number, which is uniformly expressed in the following formulas. Taking inspiration from the gated GNN [31], we set the initial vector representation of the mention node i and j to $h_i^{(0)}$=$[1; 0]^T$ and $h_j^{(0)}$=$[0; 1]^T$, respectively. While the initial vector representations of other mention nodes are set to zero.

(2) **Inter-sentence relation graph**

We construct a fully connected inter-sentence relation graph $G_2 = (V^s, E^s)$, where V^s denotes the set of sentences, and each edge $(g_i, g_j) \in E^s$ denotes adjacency (e.g., "next sentence" or "previous sentence"), coreference (from anaphora to their antecedents sentence), or discourse dependency relations (the semantic relations between text units). We add edges on all sentence node pairs to enhance the graph's connectivity.

Context-enhance sentence representation The context-enhance sentence representation embeds both semantic and augmented information of sentences into their sentence representation. To be more specific, we use sentence embedding as a basic feature to capture meaningful semantic regularities. Meanwhile, sentence relative position embedding is also used to augment the representation.

$$s_l^{i,j} = [\overline{X_l}; v_l^{i,j}], \tag{4}$$

where $\overline{X_l}$ denotes the sentence embedding of sentence s_l, and $v_l^{i,j}$ denotes the relative position embedding of sentence s_l relative to the sentence pair. These embeddings are described below:

- *Sentence embedding.*: It is represented by the average value of all word embedding representations in the sentence.
- *Sentence relative position embedding.*: In the example in Fig. 1, the relative position of sentence 2 to sentence 1 is 1, and the relative position of sentence 2 to sentence 4 is -2. Each position marker is mapped to a d_{p_s}-dimensional vector through the position embedding matrix $P \in \mathbb{R}^{2*d_{p_s}}$, where d_{p_s} is the dimension of sentence relative position embedding.

Sentence pair edge representation We feed the representations of sentence pairs into the encoder.

$$\eta_{i,j} = \text{MLP}(\text{BiLSTM}(D_{i,j})), \tag{5}$$

where $D_{i,j} = \{E(s_l^{i,j})\}_{l=1}^{L}$ denotes the representation of the sentence pair, L denotes the number of sentences in the document, and $\eta_{i,j}$ is the edge representation of the sentence pair.

Multi-layer learned sentence representation The edge representation of sentence pairs is the input to the multi-layer RNN, and the sentence vector representation is learned layer by layer:

$$g_i^{(n)} = \sigma(\sum\nolimits_{j\in N(i)} \eta_{i,j}^{(n-1)} g_j^{(n-1)} + W_g{}^{(n-1)} g_i^{(n-1)}), \tag{6}$$

where $g_i^{(n)}$ denotes the hidden vector of sentence node i at n-layer, and $W_g^{(n-1)}$ denotes a trainable parameter.

(3) **Document graph**

Through sentence number embedding, a connecting edge is established for each mention and the sentence they belong to, G_1 and G_2 are aggregated to generate a document graph G. Following the methods of [32] and [33], to capture both local and non-local interactions in the document graph, the document graph simulates information interactions through a current state transition process so that the final representation of each sentence captures the information of the entire document. During the state transition process, each node exchanges information with all its graph neighbors. After a certain number of iterations, the probability distribution of each node converges to a steady state. The document graph embedding $H^{(n)}$ is expressed as follows:

$$H^{(n)} = [H_1^{(n)}; H_2^{(n)}; \ldots; H_L^{(n)}\}, \tag{7}$$

$$H_l^{(n)} = [h_1^{(n)}; h_2^{(n)}; \ldots; h_M^{(n)}; g_l^{(n)}]. \tag{8}$$

where $H_l^{(n)}$ denotes the final representation of l-sentence in the n-iteration, which is composed of the mention vector representation $h_1^{(n)}, h_2^{(n)}, \ldots, h_M^{(n)}$ and the sentence vector representation $g_l^{(n)}$, and M is the number of mentions in the l-sentence.

As shown in Fig. 4. By default, the intra-sentence relation graph and inter-sentence relation graph only exchange information at neighbor nodes. To expedite information exchange, the size of the neighbor window is increased to allow more nodes to communicate at each state transition. For the k-th state transition, the size of the neighbor window can be expanded by k words in sequence. For example, in the second iteration, the neighbor range of $h_i^{(2)}$ in (3) is enlarged to $[h_{i-2}^{(1)}, h_{i-1}^{(1)}, h_i^{(1)}, h_{i+1}^{(1)}, h_{i+2}^{(1)}]$, and the neighbor range of $g_i^{(2)}$ in (6) is enlarged to $[g_{i-2}^{(1)}, g_{i-1}^{(1)}, g_i^{(1)}, g_{i+1}^{(1)}, g_{i+2}^{(1)}]$. With increasing iterations, the mention and sentence nodes become increasingly richer in contextual information.

Different layers of the document graph can represent the features at different abstraction levels of the l-sentence, all of which are crucial to the final representation of the l-sentence. To cover the features at all levels, we connect the representations of each level of the l-sentence to form the final representation H_l of the l-sentence.

$$H_l = [H_l^{(0)}, H_l^{(1)}, \ldots, H_l^{(n)}]. \tag{9}$$

3.2.2 Selective attention subgraph module

The selective attention subgraph module utilizes selective attention to select sentences related to the entity pair, thereby generating document subgraphs as the scope of RE. The input of this module is the final representation of each sentence and the entity pair (e_i, e_j), and the output is the document subgraph G'. The main workflow contains two steps:

Step 1: Calculating the correlation between each sentence and the entity pair; then, select the most relevant m sentences according to the correlation result.

Step 2: Aggregating m sentences into the document subgraph as the scope of RE.

For a given entity pair (e_i, e_j), the correlation between each sentence and the entity pair can be expressed as follows:

$$a_l^{i,j} = \frac{\exp(q_l^{i,j})}{\sum_{k=1}^{L} \exp(q_k^{i,j})}, \tag{10}$$

$$q_l^{i.j} = H_l^T A \varepsilon_{i,j}, \tag{11}$$

$$\varepsilon_{i,j} = MeanPooling([e_i; e_j]), \tag{12}$$

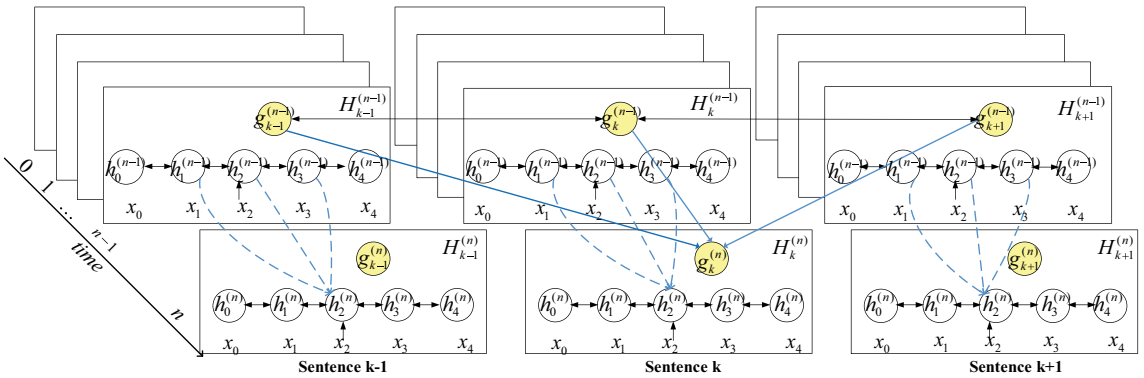


Fig. 4 State transition process of the document graph

$$e_i = \log \sum_{j=1}^{N_{e_i}} \exp(h_j). \tag{13}$$

Where $a_l^{i,j}$ denotes the normalized attention weight of each sentence vector, $q_l^{i.j}$ denotes a query-based function to measure how well the input sentence and the entity pair match, A denote a weighted diagonal matrix, and $\varepsilon_{i,j}$ denotes the query vector associated with entity pair (e_i, e_j). For an entity e_i with multiple mentions $\{h_j\}_{j=1}^{n_{e_i}}$, where n_{e_i} denotes the number of mentions for entity e_i, we apply $logsumexp$ pooling [34], a smooth version of max pooling, to obtain the hidden vector representation of the entity.

Based on the order of relevance of each sentence, m sentences are selected and spliced to obtain the subgraph $G' = \{s_k\}_{k=1}^{m}$. Finally, we define the conditional probability $P(G'|e_i, e_j, G, \theta)$ of the document subgraph through softmax layer.

$$P(G'|e_i, e_j, G, \theta) = \frac{\exp(o_{i,j})}{\sum_{u,v \in V^e} \exp\left(o_{u,v}\right)}, \tag{14}$$

$$o_{i,j} = \varepsilon_{i,j} g_{i,j}^{'} + d_{i,j}, \tag{15}$$

$$g_{i,j}^{'} = \sum_{k=1}^{m} a_k^{i,j} H_k. \tag{16}$$

where $o_{i,j}$ denotes the output vector representation of the subgraph, $d_{i,j}$ denotes the bias vector, and $g_{i,j}^{'}$ is computed as a weighted sum of all sentence vectors on the subgraph.

3.2.3 Path reasoning module

The path reasoning module explicitly models multiple associated paths with different lengths between two entities, thereby forming a unified representation of the reasoning chain, which is used to extract the intra-sentence, inter-sentence relations, and supporting evidence. The input of this module is the document subgraph G' and the entity embedding representation, and the output is the set of triples T and supporting evidence E in the document. The main workflow contains three steps:

Step 1: Constructing direct and indirect paths between the entity pair. The direct path indicates that the entity pair is in a sentence, whereas the indirect path indicates that the entity pair is not in the same sentence and that their intermediate paths need to be combined.

Step 2: Combining the direct and indirect paths to form a unified path representation.

Step 3: Using the unified path representation to predict the probability of the relation existing in the entity pair.

Path construction Reasoning paths between entities can be divided into direct paths and indirect paths. The direct path construction represents the direct relation information obtained from the subgraph G' as follows:

$$\chi_{i,j} = \left[\left[e_i^{(1)} \odot e_j^{(1)}\right]^T; \left[e_i^{(2)} \odot e_j^{(2)}\right]^T; ...; \left[e_i^{(K)} \odot e_j^{(K)}\right]^T\right]. \tag{17}$$

The indirect path represents the indirect relation information obtained from the subgraph G' using the following modified bilinear transformation:

$$f\left(\chi_{i,k}, \chi_{k,j}\right) = \sigma\left(\chi_{i,k} \odot \left(W_r \chi_{k,j}\right)\right), \tag{18}$$

where $\chi_{i,j}$ denotes the path representation between node e_i and e_j, $\odot$ denotes element-wise multiplication, and K denotes the level of the path reasoning module, which will be discussed in detail in the experimental section. W_r denotes a trainable weight matrix, σ denotes a sigmoid non-linear activation function.

Path aggregation. We use linear interpolation to aggregate direct and indirect paths as follows:

$$r_{i,j} = \alpha \chi_{i,j} + (1 - \alpha) \sum_{k \neq i,j} f\left(\chi_{i,k}, \chi_{k,j}\right), \tag{19}$$

where $\alpha \in [0, 1]$ is used to describe the relative weight between the direct and indirect paths. If the direct path provides a suitably reliable prediction, then we do not need to focus on indirect path information; moreover, the sentence on the reasoning path is marked as supporting evidence $E_{r_{i,j}}$.

Relation classification. We use a single linear layer and a sigmoid activation function to calculate the probability of relation $r_{i,j}$ on the subgraph G'.

$$P(r_{i,j}|e_i, e_j, G', \theta) = \sigma(\text{Linear}\left(r_{i,j}\right)). \tag{20}$$

In general, the relation probability $P(r_{i,j}|e_i, e_j, G, \theta)$ between the entity pairs (e_i, e_j) in the document can be calculated as follows:

$$P(r_{i,j}|e_i, e_j, G, \theta) = \sum_{G'} P(r_{i,j}|e_i, e_j, G', \theta) P(G'|e_i, e_j, G, \theta). \tag{21}$$

For a given document, we use cross-entropy for the classification loss function; that is,

$$\mathcal{L} = \sum_{i \neq j} \sum_{r_{i,j} \in \mathcal{R}} \log P(r_{i,j}|e_i, e_j, G, \theta). \tag{22}$$

Table 2 Statistics of the datasets used in the experiments

Statistics/Dataset	DocRED (human-annotated)	DocRED (distantly supervised)	CDR	GDA
Training set	3,053	101,873	500	23,353
Development set	1,000	1,000	500	5,839
Test set	1,000	1,000	500	1,000
Relations	97	97	2	2

4 Experiments

In this section, we perform experiments on three widely-used DocRE datasets to verify the effectiveness of our proposed method.

4.1 Datasets and evaluation metrics

DocRED[1] [35] is a general large-scale DocRE dataset constructed by Wikipedia and Wikidata. The dataset contains two different versions: human-annotated and distantly supervised. The human-annotated dataset includes entity mentions, entity types, relation instances, and corresponding supporting evidence, whereas the distantly supervised dataset does not contain supporting evidence. This is mainly because the construction process of the distantly supervised dataset mainly uses fine-tuned BERT to identify entities, link them to Wikidata data, and obtain relation labels through distantly supervised.

CDR[2] [36] is a human-annotated dataset in the biomedical field. This dataset represents a binary classification task that identifies induced relations between chemicals and diseases, which is of immense significance in biomedical research.

GDA[3] [37] is a large-scale distantly supervised dataset in the biomedical field. This dataset also represents a binary classification task that identifies the interaction between genes and disease concepts. It contains 29,192 documents for training and 1,000 documents for testing. Based on previous settings [15], the original training set is divided into training and development sets. The statistics of the three datasets are presented in Table 2.

Considering some relation instances present in the training and dev/test sets at the same time, the model may memorize their relation during training and perform better on the dev/test set, thereby introducing evaluation bias. Therefore, we adopted F_1 and IgnF_1 as evaluation metrics in the DocRED dataset, where IgnF_1 denotes the F_1 score that ignores the triples that appear in the training sets. The results of the DocRED test set were evaluated through CodaLab [4]. In addition, to assess the inter-sentence reasoning ability of the model on the biomedical dataset, we divided the test set into two parts based on whether an entity pair exists in the same sentence; thereafter, we reported the evaluation results of Intra-F_1 and Inter-F_1, respectively.

4.2 Baseline models

Our proposed SAPR-GNN is compared with some state-of-the-art DocRE methods. These baselines can be divided into three groups.

- **Sequence-based Models.** These models use different neural network architectures to encode sentences in a document, including **CNN** [10], **LSTM** [11], **BiLSTM** [12], and **Context-aware** [13] models. The first three models differ only in terms of the encoder used to encode the document. Context-aware models combine contextual information with an attention mechanism to predict relations between entity pairs.
- **Graph-based Models.** These models construct the document graph for reasoning. **GCNN** [14] constructs a document-level graph using coreference links and then applies a relational GCN for reasoning. **EoG** [15] is an RE model used in the biomedical field; it uses different nodes and edges to create the document graph and utilizes the edge reasoning mechanism to learn intra-sentence and inter-sentence relations. **AGGCN** [17] uses attention-guided GCN to transform the original dependency tree into a fully connected weighted graph, which is used to encode the relation between sentences. **LSR** [18] automatically generates potential document-level graphs and then utilizes refinement strategies to aggregate the relevant information to gradually achieve cross-sentence relational reasoning. **GAIN** [19] introduce graph aggregation and inference network to better cope with DocRE, which features double graphs in different granularity.
- **Transformer-based Models.** These models are fine-tuned for the DocRE dataset. **HIN** [20] introduced a hierarchical inference network to aggregate inference information from entity level to sentence level, and finally

[1] https://github.com/thunlp/DocRED

[2] https://biocreative.bioinformatics.udel.edu/media/store/files/2016/CDR_Data.zip

[3] https://bitbucket.org/alexwuhkucs/gda-extraction/get/fd4a7409365e.zip

[4] https://competitions.codalab.org/competitions/20717#results

to document level. **ATLOP** [21] introduced two novel technologies, namely adaptive thresholding and localized context pooling, to solve multi-label and multi-entity problems. **SSAN** [22] formalizes the entity structure for DocRE. It designs two transformation modules inside each self-attention block to produce attentive biases to adaptively regularize its attention flow. Its best model **SSAN+Adaptation** utilizes the distantly supervised data from DocRED to first pre-train SSAN before fine-tuning the annotated training set for better adaptation, which alleviates a distribution gap between parameters in newly introduced transformation layers and those already pre-trained ones. **DocuNet** [23] propose a document U-shaped network for DocRE. **SAIS** [24] propose to explicitly teach the model to capture relevant contexts and entity types by supervising and augmenting intermediate steps for RE. **KD-Rb** [38] proposed a novel framework for DocRE, based on knowledge distillation, axial attention, and adaptive focal loss.

4.3 Experiment settings

Experiment environment. we use the PyTorch open-source machine learning library [5] [39] to build our model. All models were trained and tested on an NVIDIA Tesla V100 GPU with 16GB of graphics memory.

Hyperparameter settings. In our experiments, we tuned the hyperparameters on the dev set. we used the Adam optimizer [40] with the cross-entropy loss function to train the model. The word embedding dimension was set to 100; the dimensions for coreference embedding, entity type embedding, word position embedding, and sentence relative position embedding were set to 20; and the hidden size of BiLSTM was 256. We selected the nonlinear activation functions in $ReLU$ and $tanh$ with the learning rate $\lambda \in \{0.1, 0.01, 0.001, 0.0001\}$ and dropout rate in $\{0.1, 0.2, 0.3, 0.4, 0.5\}$. The early stopping mechanism was used to determine the best training epoch [41]. We tuned the hyperparameters on the dev set. With the $ReLU$ activation function and the learning rate and the dropout rate of 0.001 and 0.5, respectively, our model achieved the best performance. In addition, The dev set of the human-annotated DocRED dataset statistics indicates that the average number of supporting evidence for relation instances in each document is 2.6. Therefore, we set the maximum number of sentences contained in the subgraph to 3. After calculating the attention weights (10), we selected the top three sentences with the largest weights to generate a subgraph.

[5] https://pytorch.org

We also used different pre-trained language models as the encoder in the experiment. For the DocRED dataset, we utilized the BERT-base [42] or Roberta-large models[43]; For the CDR and GDA datasets, we utilized the SciBERT-base model [44], which is pre-trained on the scientific publication corpora. The learning rate λ was selected from $\{1e-5, 2e-5, 3e-5, 4e-5, 5e-5\}$. The learning rate for fine-tuning BERT is 5e-5, that for fine-tuning SciBERT or distant pretrain is 2e-5, and the dropout rate of 0.1, our model obtained the best performance.

4.4 Extraction result comparison

Results on DocRED human-annotated dataset. The experiment results on DocRED are reported in Table 3.

1) For Sequence-based models, we can observe that Context-aware consistently outperforms all baselines on the dev set. The results demonstrate that rich contextual information is essential; however, this method ignores whether contextual relation participates in the relational reasoning process of the entity pair. Therefore, Context-aware could not significantly outperform other models on the test set, and our model outperformed all sequence-based models consistently. Compared with BiLSTM, GloVe+SAPR-GNN (#layers = 3) yielded a 5.49% improvement in F_1 on the test set.
2) For Graph-based models, Our model consistently obtains the best score than baselines. Compared with static graph models, GloVe+SAPR-GNN (#layers = 3) achieves an improvement of F_1 on the test set by 4.93% (GCNN) and 4.73% (EoG). The comparison results demonstrate that the static graph model cannot capture the complex interactions in the document. Compared with attention mechanism-based graph models, GloVe+SAPR-GNN (#layers = 3) achieves an improvement of F_1 on the test set by 5.10% (AGGCN). Compared with GloVe+LSR/GloVe+GAIN, GloVe+SAPR-GNN (#layers = 3) achieves an improvement of F_1 on the test set by 2.37%/1.47%. That means that our model can make full use of sentences related to the entity pair and combine path reasoning for better RE in the document.
3) For Transformer-based models, BERT+SAPR-GNN (#layers = 3) outperforms most models by an improvement of F_1 on the test set, which further indicates that SAPR-GNN can offer consistent and robust improvements. We can also observe that compared with GloVe+SAPR-GNN (#layers = 3), BERT+SAPR-GNN (#layers = 3) achieves an improvement of 6.25%. The comparison results show that using a transformer as a context encoder can sig-

nificantly improve the performance of RE. When using Roberta-large as the encoder, the Roberta+SAPR-GNN (#layers=3) model achieved an F_1 score of 63.90.

As shown in Table 3, distantly supervised data can improve the performance of DocRE. We conducted experiments to leverage the distantly supervised data to improve our model performance. Compared with RoBERTa+SAPR-GNN (#layer=3), SAPR-GNN+Adaptation achieves an improvement of F_1 on the test set by 1.84%. Besides, we also found that SSAN+Adaptation and KD-Rb also have certain competitive advantages. This is mainly because SSAN incorporates these structural dependencies within the standard self-attention mechanism and throughout the overall encoding stage. KD-Rb uses knowledge distillation to overcome the differences between human-annotated data and distantly supervised data. Moreover, KD-Rb also tackles the under-explored class imbalance problem and the two-hop logical reasoning problem. We plan to integrate the advantages of these works in our future work to improve the effectiveness of our model.

Table 3 Performance of different baseline models and SAPR-GNN on the DocRED human-annotated dataset, where the numbers in boldface indicate the best results

Model	Dev		Test	
	IgnF_1	F_1	IgnF_1	F_1
CNN* [35]	41.58	43.45	40.33	42.26
LSTM* [35]	48.44	50.68	47.71	50.07
BiLSTM* [35]	48.87	50.94	48.78	51.06
Context-Aware* [35]	48.94	51.09	48.40	50.70
GCNN‡ [14]	46.22	51.52	49.59	51.62
EoG‡ [15]	45.94	52.15	49.48	51.82
AGGCN‡ [17]	46.29	52.47	48.89	51.45
GloVe+LSR* [18]	48.82	55.17	52.15	54.18
GloVe+GAIN* [19]	53.05	55.29	52.66	55.08
GloVe+SAPR-GNN (#layers = 1)	49.20	53.50	50.25	52.45
GloVe+SAPR-GNN (#layers = 2)	52.20	55.90	51.95	54.80
GloVe+SAPR-GNN (#layers = 3)	**54.20**	**57.40**	**53.35**	**56.55**
BERT+HIN* [20]	54.29	56.31	53.70	55.60
BERT+SSAN* [22]	57.03	59.19	55.84	58.16
BERT+LSR* [18]	52.43	59.00	56.97	59.05
BERT+GAIN* [19]	59.14	61.22	59.00	61.24
BERT+ATLOP* [21]	59.22	61.09	59.31	61.30
BERT+DocuNet* [23]	59.86	61.83	59.93	61.86
BERT+SAIS* [24]	59.98	62.96	**60.96**	62.77
BERT+SAPR-GNN (#layers = 1)	55.27	59.39	54.15	58.26
BERT+SAPR-GNN (#layers = 2)	58.21	61.92	57.76	60.65
BERT+SAPR-GNN (#layers = 3)	**60.22**	**63.09**	60.70	**62.80**
RoBERTa+SSAN* [22]	60.25	62.08	59.47	61.42
RoBERTa+ATLOP* [21]	61.32	63.18	61.39	63.40
RoBERTa+DocuNet* [23]	62.23	64.12	62.39	64.55
RoBERTa+SAIS* [24]	**62.23**	**65.17**	**63.44**	**65.11**
RoBERTa+SAPR-GNN (#layers = 1)	56.46	60.26	55.39	59.31
RoBERTa+SAPR-GNN (#layers = 2)	59.47	62.70	59.00	61.78
RoBERTa+SAPR-GNN (#layers = 3)	61.45	64.06	61.80	63.90
SSAN+Adaptation* [22]	63.76	65.69	63.78	65.92
KD-Rb* [38]	**65.27**	**67.12**	**65.24**	**67.28**
SAPR-GNN+Adaptation	63.26	65.33	63.39	65.74

* indicates that the results are reported from their original papers. ‡ indicates that the results are reported from [18]

Results on biomedical datasets. Table 4 shows the experiment results on biomedical datasets CDR and GDA. Our proposed SciBERT+SAPR-GNN (#layers = 3) consistently obtains the best score than most baseline models. The comparison results demonstrate the applicability and generality of our model. Similar performance gains are also observed in SAIS, this is mainly because SAIS captures textual contexts and entity types information for RE similar to our model. The difference is SAIS extracts relations of better quality due to more effective supervision and retrieves the corresponding supporting evidence more accurately due to Pooled Evidence Retrieval (PER) and Fine-grained Evidence Retrieval (FER), where PER distinguishes entity pairs with and without valid supporting sentences and FER output more interpretable evidence unique to each valid relation of an entity pair. We plan to exploit these modules in our further work that improves the effect of RE.

We also observe that most models trained with CDR data were superior to that trained with GDA data for inter-sentence RE, which attribute this behavior to the fact that there are only a few inter-sentence relations in the GDA dataset, which led to insufficient training of these models. However, another phenomenon is that the DHG method outperforms all baseline models in the inter-sentence setting in GDA. This is mainly because DHG adds entity-to-entity complement edges in the second-tiar layer, this type of edge can prevent having disconnected graphs and enhance the multi-hop reasoning ability.

Effect of layer number. The number of layers represents the reasoning ability of our models. A K-layer version can infer K-hop relations. To explore the impact of the number of layers, we also compare our models with different numbers of layers. From Tables 3 and 4, we could see that on all three datasets, the 3-layer version achieves the best, indicating considering more hops in reasoning leads to better performance. However, neither the shallow model nor the deep model works very well. One possible reason is that only collecting information from the nearest neighbor nodes is not enough to identify the relation between two entities. In contrast, when the layer number is above 3, any two nodes in the same graph are accessible, which may introduce redundant information and hinder the inference.

4.5 Supporting evidence prediction

In this section, we further explore the performance of our model for supporting evidence prediction on the DocRED dev set. On the one hand, supporting evidence can provide better explainability for predicted relation instances. On the other hand, identifying supporting evidence and reasoning relational facts from the text are naturally dual tasks with potential mutual enhancement.

Moreover, most earlier approaches are not capable of supporting evidence prediction despite the explainability supporting evidence prediction can increase the predictive performance of the model. In contrast, BERT+SAPR-GNN (#layers = 3) establishes a new state-of-the-art result on supporting evidence prediction. As shown in Table 5, we also observe that GloVe+SAPR-GNN is significantly superior to the heuristic predictor and neural predictor. The heuristic predictor considers all sentences containing the head or tail entity as supporting evidence, introducing substantial irrelevant supporting evidence, which reduces the accuracy of the supporting evidence prediction. The neural predictor first converts sentences into input representations by concatenating word and position embedding. It then feeds the representations into a BiLSTM encoder for contextual representation. Finally, the output of the BiLSTM is concatenated

Table 4 Results were obtained with CDR and GDA test datasets, where the numbers in boldface indicate the best results

Model	CDR			GDA		
	F_1	Intra-F_1	Inter-F_1	F_1	Intra-F_1	Inter-F_1
EoG* [15]	63.6	68.2	50.9	81.5	85.2	49.3
LSR* [18]	61.2	66.2	50.3	79.6	83.1	49.6
LSR w/o MDP Nodes* [18]	64.8	68.9	53.1	82.2	85.4	51.1
DHG [28]*	65.9	70.1	54.6	83.1	85.6	**58.8**
SAIS [24]*	**79.0**	–	–	**87.1**	–	–
PubMed/Random+SAPR-GNN (#layers = 1)	57.2	61.4	46.3	75.1	78.5	44.9
PubMed/Random+SAPR-GNN (#layers = 2)	59.2	63.0	48.2	77.3	80.3	46.2
PubMed/Random+SAPR-GNN (#layers = 3)	61.6	65.4	50.8	79.4	82.2	48.3
SciBERT+SAPR-GNN (#layers = 1)	63.2	67.1	52.4	81.7	84.2	51.5
SciBERT+SAPR-GNN (#layers = 2)	64.3	68.2	52.9	82.7	85.9	51.9
SciBERT+SAPR-GNN (#layers = 3)	67.8	**72.4**	**56.1**	83.9	**88.4**	53.2

* indicates that the results are reported from their original papers. We also present the intra-F_1 and inter-F_1 for further analysis

Table 5 Performance of supporting evidence prediction in F_1 measurement, where the numbers in boldface indicate the best results

Model	Dev	Test
Heuristic predictor [35]	36.21	36.76
Neural predictor [35]	44.07	43.83
GloVe+SAPR-GNN (#layers = 1)	45.85	44.25
GloVe+SAPR-GNN (#layers = 2)	47.38	46.02
GloVe+SAPR-GNN (#layers = 3)	**49.96**	**48.75**
BERT+E2GRE [45]	47.12	–
BERT+Eider [46]	50.71	51.27
BERT+SAIS [24]	53.70	52.88
BERT+SAPR-GNN (#layers = 1)	51.25	50.25
BERT+SAPR-GNN (#layers = 2)	53.36	52.63
BERT+SAPR-GNN (#layers = 3)	**55.48**	**54.75**

at the first and last positions with a trainable relation embedding to obtain the representation of a sentence, which is used to predict whether the sentence is adopted as supporting evidence for the given relation instance. The neural predictor ignores the role of the intermediate path in the path reasoning process, leading to a lower recall in the supporting evidence prediction. Our model SAPR-GNN can reduce the influence of irrelevant sentences through selective attention and path reasoning mechanism to obtain an effective reasoning path. In this manner, a reliable balance is achieved between precision and recall, and the best F_1 score is obtained. Furthermore, based on the order in which sentences appear on the reasoning path, the order of supporting evidence extraction can be determined, thus providing better explainability for DocRE.

5 Analysis and discussion

In this section, we use the DocRED dev set to analyze and discuss model complexity, the impact of different supporting evidence, ablation experiments, case studies, and visualization analysis of selective attention. The experiments were conducted with the following five research objectives:

- Evaluating the computational complexity of our proposed SAPR-GNN.
- Analyzing the extraction performance of the model under different numbers and types of supporting evidence.
- Analyzing the impact of the network structure and features of our proposed SAPR-GNN on the model extraction performance.
- Showing the actual effect of our proposed SAPR-GNN and other baseline models in real cases.
- Quantifying the distribution of selective attention weights and providing explainable analysis for the selection of supporting evidence.

5.1 Complexity analysis

In this subsection, the complexity analysis of our proposed SAPR-GNN is provided. The computational complexity of our proposed SAPR-GNN is the sum of the three submodules. Suppose there are m sentences in a document and that each sentence has n_i mentions. In the GNN–RNN module, all mention nodes are fully connected, and their time complexity is $\sum_{i=1}^{m} n_i^2$. Additionally, all sentence nodes are fully connected, and their time complexity is m^2. In the selective attention module, the upper bound on the number of sentences contained in the generated subgraph is m. In the path reasoning module, suppose there is $\bar{n}$ mentions on average in each sentence of the document and $m\bar{n}$ mentions in the subgraph, and that the average length of path reasoning is $\frac{(m\bar{n}-1)}{2}$. The overall complexity is then $\sum_{i=1}^{m} n_i^2 + m^2 + m + \frac{(m\bar{n}-1)}{2}$. Assuming the number of iterations of the model is a constant j, the computational complexity of our proposed SAPR-GNN is:

$$\mathrm{O}(j(\sum_{i=1}^{m} n_i^2 + m^2 + m + \frac{(m\bar{n}-1)}{2})) = \mathrm{O}(m\bar{n}^2 + m^2). \quad (23)$$

According to the statistics on the DocRED dev set, we found that the average number of sentences in each document m is 8.1 and that the average number of mentions in each sentence $\bar{n}$ is 3.5. Therefore, the average computational complexity is $\mathrm{O}(m\bar{n}^2 + m^2)$, which is only related to the number of sentences in the document and the average number of mentions in the sentence.

To compare the computational cost of SAPR-GNN and baseline models (SSAN/ATLOP/DocuNet), we conduct an experiment about the time cost. Experiments show that training and testing our model requires 6h on a single Tesla V100 GPU, while baseline models require 9h∼11h on the same GPU, which shows SAPR-GNN runs 1.5∼1.8 faster than baseline models, it has a significant reduction in computational cost. This is mainly because SAPR-GNN employs a selective attention subgraph module to select the most relevant sentences, which can filter out meaningless sentences, and utilize these selected sentences to predict the relation. However, the computational complexity of these baseline models is similar to that of BERT, and its value is $o\left(seq\_len\hat{}2\right)$ (seq_len is the length of the entire document sequence). Besides, we note that when SAIS is uncertain about its original predictions, it applies evidence-based data

augmentation with ensemble inference to boost the model's RE performance after the training of SAIS. However, SAIS sets a low rejection rate so that data augmentation on a small rejected set reduces the computational cost. Therefore, SAIS and our model have certain advantages in terms of computational cost.

5.2 Supporting evidence analysis

In this section, the analysis of the supporting evidence is provided. It should be noted that when the RE model predicts a wrong relation, it is not possible to directly determine sentences that were used as supporting evidence. In this case, computing precision is infeasible. Therefore, we used the recall rate, as indicated in the following two subsections, to measure the extraction performance of our proposed SAPR-GNN under different supporting evidence.

Different numbers of supporting evidence. To analyze how different numbers of supporting evidence affect the performance of our proposed model, we conducted recall experiments on relation instances with different numbers of supporting evidence. The experiment results are shown in Fig. 5.

The recall of 2-3 and 4-5 seems lower than the recall of 0-1 for most models. This is mainly because supporting evidence is positively correlated with the number of hops in path reasoning. The scope of path reasoning increases with increasing supporting evidence, which leads to an increase in the difficulty of reasoning. We also observed that when the number of supporting evidence items exceeds seven, the performance of other models decreases. This is mainly because

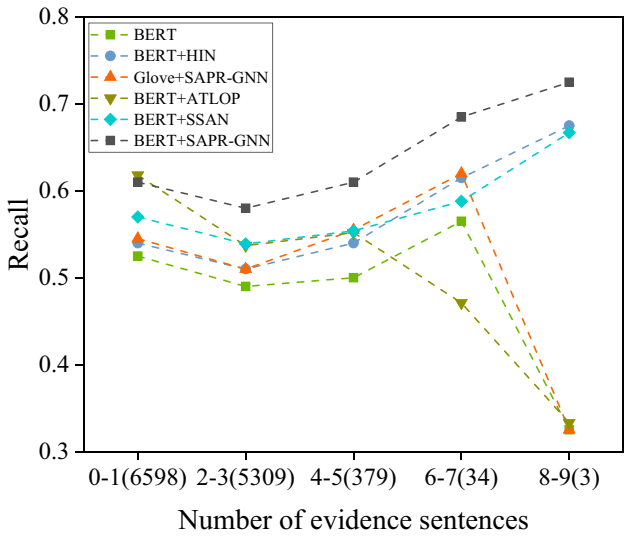


Fig. 5 Recall of relation instances under different numbers of supporting evidence. The number in brackets is the number of relation instances with the given number of supporting evidence

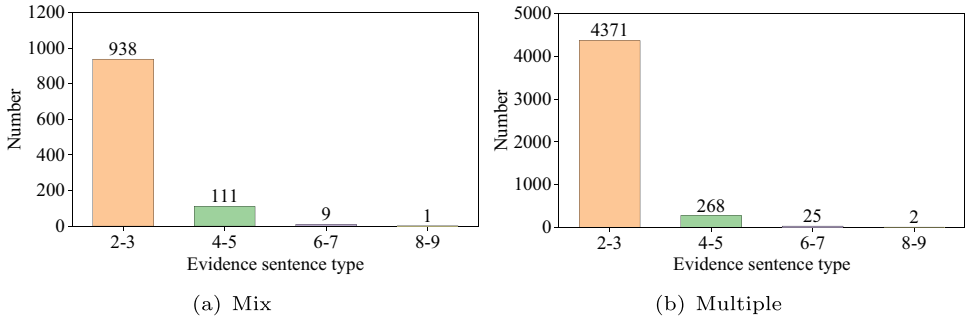
there are very few samples with more than seven supporting evidence items on the dev set, which affects the performance of other models. However, the performance of BERT+SSAN, BERT+HIN, and BERT+SAPR-GNN continues to increase gradually, and BERT+SAPR-GNN consistently outperforms the two other models. With an increasing number of supporting evidence, this improvement becomes larger. This is mainly because our proposed SAPR-GNN introduces cross-sentence relation path reasoning, and always performs the best on inter-sentence relation prediction (more than one evidence sentence), thus improving the DocRE performance.

Different types of supporting evidence. To investigate the difficulty of synthesizing information from different types of supporting evidence, we divided the 12,323 relation instances on the dev set into three different types:

- **Single relation instance**. In 6,111 relation instances, only one supporting evidence is needed to predict the relation instance.
- **Mix relation instance**. In 1,059 relation instances, multiple supporting evidences are needed to predict the relation instance, and the entity pair co-occur in at least one supporting evidence.
- **Multiple relation instance**. In 4,666 relation instances, multiple supporting evidences are needed to predict the relation instance, and the entity pair do not co-occur in any supporting evidence. The relation between entity pairs can only be extracted from multiple supporting evidence.

The distribution of the number of supporting evidence in Mix and Multiple types is shown in Fig. 6. In Fig. 6a, 88.57% of the relation instances require 2-3 supporting evidence, and 10.48% of the relation instances need 4-5 supporting evidence. In Fig. 6b, 93.68% of the relation instances need 2-3 supporting evidence, and 5.74% of the relation instances need 4-5 supporting evidence. The number of relation instances with six or more supporting evidence is relatively small; therefore, it is more challenging to extract relation instances with more than six supporting evidence.

We also compared the recall of our proposed SAPR-GNN and the baseline model on DocRE under three different types of supporting evidence. The experiment results are presented in Table 6. Although multiple supporting evidence can provide additional supplementary information, the extraction performance is not comparable to that achieved with single supporting evidence. It is challenging to integrate global information among multiple supporting evidence. Compared with BiLSTM, both GloVe+SAPR-GNN and BERT+SAPR-GNN achieved improvement under different types of supporting evidence. We also found that BERT+SAPR-GNN (#layers = 3) achieved significantly bet-



(a) Mix (b) Multiple

Fig. 6 Distribution of the number of supporting evidence in the mix and multiple types

ter results than BERT+SSAN and BERT+ATLOP under multiple relation instances. This is mainly because our model can perform path reasoning, transfer information between multiple sentences, and facilitate intra-sentence and inter-sentence RE tasks.

5.3 Ablation study

To study the contributions of different modules in our model, we run an ablation study on the DocRED dev set. We show the results of the ablation study in Table 7.

In terms of network structure, when we remove the G_1, we initialize a mention node with (3) but replace $h_i^{(n)}$ with $h_i^{(0)}$. Without G_1, the performance of BERT+SAPR-GNN sharply drops by 2.08% and 2.23% in the IgnF_1 and F_1 scores on the dev set. This drop shows that G_1 plays a vital role in capturing interactions among mentions belonging to the same sentence. when we remove the G_2. In detail, we initialize a mention node with (6) but replace $g_i^{(n)}$ with $g_i^{(0)}$. Without G_2, the performance of BERT+SAPR-GNN sharply drops by 1.92% and 2.06% in the IgnF_1 and F_1 scores on the dev set. This drop shows that G_2 plays a vital role in capturing interactions among sentences belonging to the same document.

Table 6 Recall of models on relation instances with different types of supporting evidence, where the numbers in boldface indicate the best results

Model	Single	Mix	Multiple
BiLSTM [35]	51.10	49.40	46.60
BERT+SSAN [22]	58.83	62.51	52.08
BERT+ATLOP [21]	**61.81**	**64.87**	51.24
GloVe+SAPR-GNN (#layers = 1)	51.50	50.25	47.35
GloVe+SAPR-GNN (#layers = 2)	53.42	51.38	48.63
GloVe+SAPR-GNN (#layers = 3)	57.22	55.25	52.16
BERT+SAPR-GNN (#layers = 1)	57.55	55.50	52.92
BERT+SAPR-GNN (#layers = 2)	59.36	57.49	54.36
BERT+SAPR-GNN (#layers = 3)	61.03	59.25	**56.32**

We also observe that the contribution of G_1 to document graph representation is more, this is mainly because there are 52% intra-sentence relations on the dev set. Next, we remove the selective attention subgraph module, and the performance of BERT+SAPR-GNN decreases on the dev set, as indicated by a reduction of 1.84% and 1.97% in the IgnF_1 and F_1 score, respectively. This is mainly because the selective attention subgraph module can reduce the influence of irrelevant sentences. Moreover, taking away the path reasoning module leads to 5.80% and 5.97% in the IgnF_1 and F_1 score decrease. This implies that the path reasoning module is available to infer the relation between entities.

In terms of features, the entity type offers the most significant effect, mainly because it directly affects the relation type. The sentence number and relative position are crucial for identifying cross-sentence relations and extracting supporting evidence from documents; in addition, coreference and word position are essential for synthesizing information from multiple mentions. When we removed all augment representation, the IgnF_1 and F_1 of BERT+SAPR-GNN decreased by 4.05% and 3.66%, respectively, on the dev set. This

Table 7 Ablation study of BERT+SAPR-GNN on the DocRED dev set. We exclude one module or feature in the model at a time, where the numbers in boldface indicate the best results

Setting	IgnF_1	F_1
BERT+SAPR-GNN	**60.22**	**63.09**
w/o G_1 module	58.14	60.86
w/o G_2 module	58.30	61.03
w/o Selective attention subgraph module	58.38	61.12
w/o Path reasoning module	54.42	57.12
w/o Entity type	58.21	60.81
w/o Sentence number	58.70	61.56
w/o Sentence relative position	58.95	61.72
w/o Coreference	59.26	62.57
w/o Word position	59.45	62.84
w/o Augment representation	56.17	59.43

implies that the participation of multi-channel information can enhance the expression ability of BERT+SAPR-GNN at the entity mention and sentence level.

5.4 Case study

In this subsection, the case study of our proposed SAPR-GNN is provided. Our goal here is to predict the relationship between the entity pair (*Japan*, *World War II*). The results are reported in Fig. 7, we found that:

1) In BERT+SAPR-GNN (#layers = 1), the mention interacts with local mentions in the same sentence, such as "World War II" and "Lark Force", "World War II" and "Australian Army". The experimental result shows that information is propagated locally at this step. In BERT+SAPR-GNN (#layers = 2), the mention interacts with several non-local mentions, such as "Australian Army" and "Lark Force", "World War II" and "Imperial Japanese Army", "Imperial Japanese Army" and "Japan". The experimental result shows that information starts to propagate globally at this step. Through such intra- and inter-sentence information interaction, the relation of the entity pair (*Japan*, *World War II*) can be predicted as "participant of" by BERT+SAPR-GNN (#layers = 3), which is denoted by P1344.
2) Context-aware and AGGCN methods only predicted the relations P607 and P17; thus, they failed to predict the entity pair (*Japan*, *World War II*) with the relation P1344. LSR and BERT+SAPR-GNN can predict the relation accurately. This shows that the iterative refinement strategy of LSR and multi-hop reasoning of SAPR-GNN can help the model perform better relational reasoning.
3) LSR assigns a relatively high score to "New Ireland". There is no relation between (*New Ireland*, *World War II*). SAPR-GNN can make full use of the relevant sen-

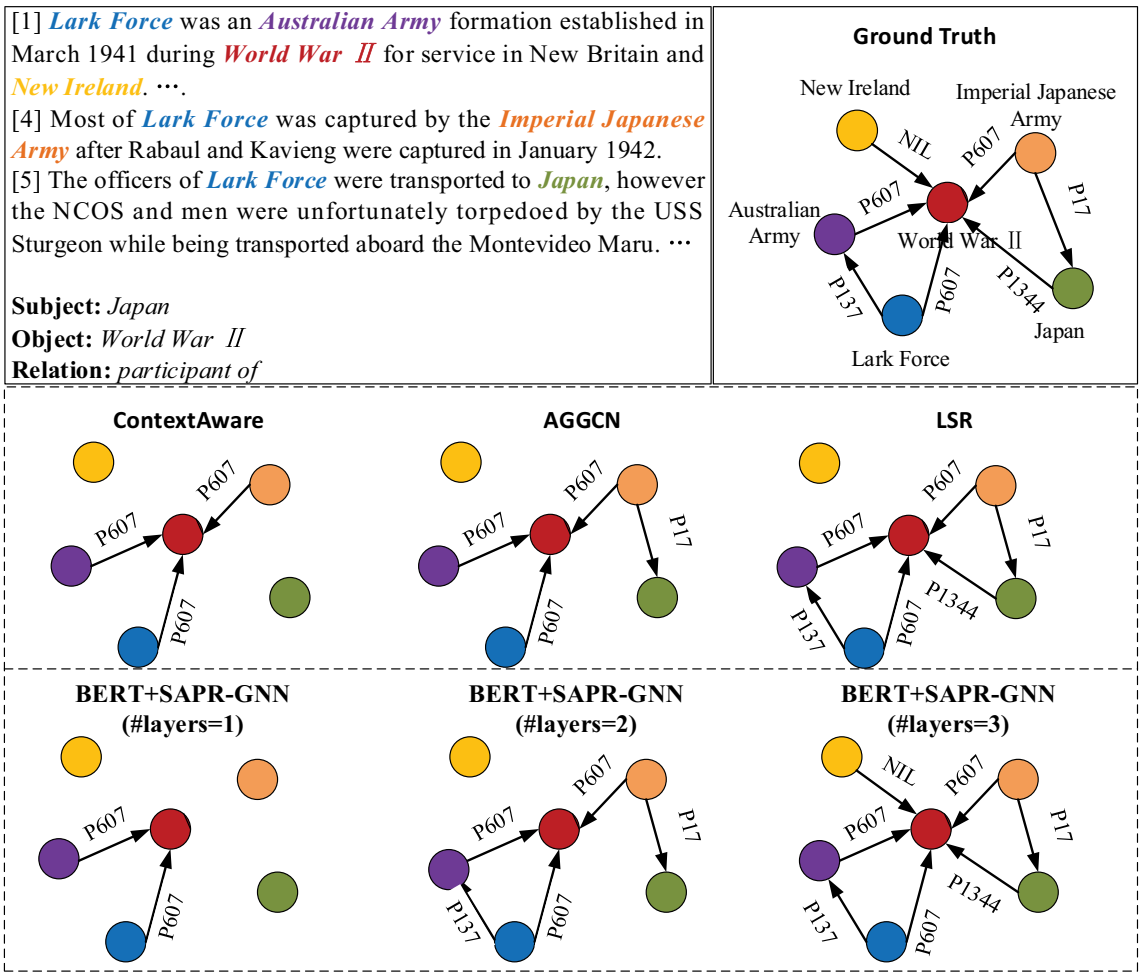
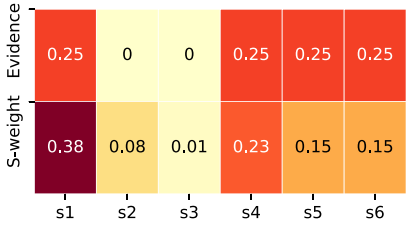


Fig. 7 Case study of an example from the DocRED dev set, where P17 denotes "country," P137 denotes "operator," P607 denotes "conflict," and P1344 denotes "participant of." In addition, we visualize the reasoning process of BERT+SAPR-GNN to predict the relation of the entity pair (*Japan*, *World War II*)

[1] Lark Force was an Australian Army formation established in March 1941 during World War II for service in New Britain and New Ireland. [2] Under the command of Lieutenant Colonel John Scanlan, it was raised in Australia and deployed to Rabaul and Kavieng, aboard SS Katoomba, MV Neptuna and HMAT Zealandia, to defend their strategically important harbours and airfields. [3] The objective of the force, was to maintain a forward air observation line as long as possible and to make the enemy fight for this line rather than abandon it at the first threat as the force was considered too small to withstand any invasion. [4] Most of Lark Force was captured by the Imperial Japanese Army after Rabaul and Kavieng were captured in January 1942. [5] The officers of Lark Force were transported to Japan, however the NCOs and men were unfortunately torpedoed by the USS Sturgeon while being transported aboard the Montevideo Maru. [6] Only a handful of the Japanese rew were rescued, with none of the between 1,050 and 1,053 prisoners aboard surviving as they were still locked below deck.

Ground Truth: participant of
The Prediction of SAPR-GNN: participant of

(a) Document



(b) Visualization

Fig. 8 (a) Document containing six sentences. Words in red are target entities, words in blue are non-target entities, and the evidence sentences are underlined. (b) Visual results of selective attention on the DocRED development set; the deeper color indicates a greater weight

tences in the document and incorporate the multi-path reasoning mechanism to eliminate the existence of such empty relations.

5.5 Visualization analysis of selective attention

Figure 8 shows the visualization analysis of selective attention. As shown in Fig. 8a, the red word in the figure is the target entity, the blue word is the non-target entity, and the underlined sentence is the supporting evidence. It should be noted that since only supporting evidence is marked on the dev set, the attention weight is not marked. Inspired by Li et al. [47], it is assumed that the weight of the four supporting evidence is 0.25, respectively, and then it is compared with the first four sentences with the largest attention weight generated by our model to judge whether the real supporting evidence is consistent with the predicted results. As shown in Fig. 8b, The "Evidence" row shows the actual attention weights of sentences in the document, and the "S-weight" row shows the attention weights generated by our model.

In Fig. 8, we can observe that the supporting evidence predicted by our model is consistent with the real supporting evidence. In addition, we can observe the following: 1) Our model pays more attention to information sentences, such as sentences containing target entities, bridging entities, and sentences representing the relation between entities. For example, when predicting the relation of the entity pair (*Japan*, *World War II*), our model pays more attention to sentences 1, 4, 5, and 6. These sentences are essential for the relation "participant of." 2) The visualization results not only confirm the effectiveness of supporting evidence prediction but also reveal the explainability of our proposed SAPR-GNN.

6 Conclusion

In this paper, we proposed a novel DocRE method. The method first utilizes GNN–RNN module to realize hierarchical document modeling and information interaction. Then, selective attention is introduced to generate document subgraphs as the scope of RE. Finally, path reasoning is introduced to infer the relation between entities and provide supporting evidence for each relation instance. Our model enhances the explainability of the extraction results. The extensive experiment results conducted on several public DocRE datasets demonstrate that our proposed approach outperforms most existing methods. Further analysis shows selective attention and path reasoning are able to discover more accurate inter-sentence relations and supporting evidence.

We also performed error analysis on the experiment results and found that a more reasonable method may be necessary to exclude irrelevant sentences, instead of simply setting a fixed length constraint, such as three, for all instances in the DocRED dataset. This work will be carried out in our subsequent study. Additionally, we plan to improve the performance of DocRE in the following aspects: 1) we will consider the impact of more fine-grained supporting evidence and supporting evidence extraction order on DocRE; 2) we will investigate how to avoid error transmission between RE and supporting evidence prediction; 3) we will explore the combination of relational paths from plain text and external knowledge bases.

Acknowledgements The authors would like to thank the anonymous reviewers for their encouragement and helpful comments. The paper is supported by the National Natural Science Foundation of China (Grant No. 62306007), the National Key R&D Program of China (Grant

No. 2021YFB3900601), and the Natural Science Foundation of Anhui Province (Grant No. 2008085QF305).

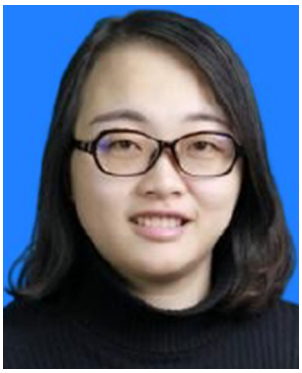
Declarations

Conflicts of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Distiawan B, Weikum G, Qi J, Zhang R (2019) Neural relation extraction for knowledge base enrichment. In: Proceedings of the 57th annual meeting of the association for computational linguistics. https://doi.org/10.18653/v1/p19-1023
2. Yu M, Yin W, Hasan KS, dos Santos C, Xiang B, Zhou B (2017) Improved neural relation detection for knowledge base question answering. In: Proceedings of the 55th annual meeting of the association for computational linguistics. https://doi.org/10.18653/v1/P17-1053
3. Lai T, Cheng L, Wang D, Ye H, Zhang W (2022) Rman: Relational multi-head attention neural network for joint extraction of entities and relations. Appl Intell 52(3):3132–3142. https://doi.org/10.1007/s10489-021-02600-2
4. Li X, Li Y, Yang J, Liu H, Hu P (2022) A relation aware embedding mechanism for relation extraction. Appl Intell, pp 1–10. https://doi.org/10.1007/s10489-021-02699-3
5. Christopoulou F, Miwa M, Ananiadou S (2018) A walk-based model on entity graphs for relation extraction. In: Proceedings of the 56th annual meeting of the association for computational linguistics. https://doi.org/10.18653/v1/P18-2014
6. Zhu H, Lin Y, Liu Z, Fu J, Chua T-s, Sun M (2019) Graph neural networks with generated parameters for relation extraction. In: Proceedings of the 57th conference of the association for computational linguistics. https://doi.org/10.18653/v1/p19-1128
7. Wang H, Qin K, Lu G, Luo G, Liu G (2020) Direction-sensitive relation extraction using bi-sdp attention model. Knowl Based Syst, pp 105928. https://doi.org/10.1016/j.knosys.2020.105928
8. Hang T, Feng J, Wu Y, Yan L, Wang Y (2021) Joint extraction of entities and overlapping relations using source-target entity labeling. Expert Syst Appl 177:114853. https://doi.org/10.1016/j.eswa.2021.114853
9. Hang T, Feng J, Yan L, Wang Y, Lu J (2022) Joint extraction of entities and relations using multi-label tagging and relational alignment. Neural Comput Appl 34(8):6397–6412. https://doi.org/10.1007/s00521-021-06685-1
10. Zeng D, Liu K, Lai S, Zhou G, Zhao J (2014) Relation classification via convolutional deep neural network. In: Proceedings of the 4th international conference on learning representations
11. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780
12. Cai R, Zhang X, Wang H (2016) Bidirectional recurrent convolutional neural network for relation classification. https://doi.org/10.18653/v1/p16-1072
13. Sorokin D, Gurevych I (2017) Context-Aware representations for knowledge base relation extraction. In: Proceedings of the 2017 conference on empirical methods in natural language processing
14. Sahu SK, Christopoulou F, Miwa M, Ananiadou S (2019) Inter-sentence relation extraction with document-level graph convolutional neural network. In: Proceedings of the 57th annual meeting of the association for computational linguistics. https://doi.org/10.18653/v1/p19-1423
15. Christopoulou F, Miwa M, Ananiadou S (2019) 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. https://doi.org/10.18653/v1/D19-1498
16. Wang D, Hu W, Cao E, Sun W (2020) Global-to-local neural networks for document-level relation extraction. In: Proceedings of the 2020 conference on empirical methods in natural language processing. https://doi.org/10.18653/v1/2020.emnlp-main.303
17. Guo Z, Zhang Y, Lu W (2019) Attention guided graph convolutional networks for relation extraction. In: Proceedings of the 57th annual meeting of the association for computational linguistics. https://doi.org/10.18653/v1/p19-1024
18. Nan G, Guo Z, Sekulic I, Lu W (2020) Reasoning with latent structure refinement for document-Level relation extraction. In: Proceedings of the 58th annual meeting of the association for computational linguistics. https://doi.org/10.18653/v1/2020.acl-main.141
19. Zeng S, Xu R, Chang B, Li L (2020) Double graph based reasoning for document-level relation extraction. In: Proceedings of the 2020 conference on empirical methods in natural language processing. https://doi.org/10.18653/v1/2020.emnlp-main.127
20. Tang H, Cao Y, Zhang Z, Cao J, Fang F, Wang S, Yin P (2020) HIN: hierarchical inference network for document-level relation extraction. In: Pacific-Asia conference on knowledge discovery and data mining. https://doi.org/10.1007/978-3-030-47426-3_16
21. Zhou W, Huang K, Ma T, Huang J (2021) Document-level relation extraction with adaptive thresholding and localized context pooling. In: Proceedings of the 35th AAAI conference on artificial intelligence
22. Xu B, Wang Q, Lyu Y, Zhu Y, Mao Z (2021) Entity structure within and throughout: modeling mention dependencies for document-level relation extraction. In: Proceedings of the 35th AAAI conference on artificial intelligence
23. Zhang N, Chen X, Xie X, Deng S, Tan C, Chen M, Huang F, Si L, Chen H, Center HI (2021) Document-level relation extraction as semantic segmentation. In: Proceedings of the 30th international joint conference on artificial intelligence. https://doi.org/10.24963/ijcai.2021/5517
24. Xiao Y, Zhang Z, Mao Y, Yang C, Han, J (2022) SAIS: supervising and augmenting intermediate steps for document-level relation extraction. https://doi.org/10.18653/v1/2022.naacl-main.171
25. Quirk C, Poon H (2017) Distant supervision for relation extraction beyond the sentence boundary. In: Proceedings of the 15th conference of the european chapter of the association for computational linguistics. https://doi.org/10.18653/v1/e17-1110
26. Peng N, Poon H, Quirk C, Toutanova K, Yih W-t (2017) Cross-sentence n-ary relation extraction with graph lstms. Trans Assoc Comput Linguistics 5:101–115. https://doi.org/10.1162/tacl_a_00049
27. Song L, Zhang Y, Wang Z, Gildea D (2018) N-ary relation extraction using graph state LSTMs. In: Proceedings of the 2018 conference on empirical methods in natural language processing. https://doi.org/10.18653/v1/d18-1246
28. Zhang Z, Yu B, Shu X, Liu T, Tang H, Yubin W, Guo L (2020) Document-level relation extraction with dual-tier heterogeneous graph. In: Proceedings of the 28th international conference on computational linguistics. https://doi.org/10.18653/v1/2020.coling-main.143
29. Pennington J, Socher R, Manning CD (2014) Glove: global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing. https://doi.org/10.3115/v1/d14-1162
30. Chiu B, Crichton G, Korhonen A, Pyysalo S (2016) How to train good word embeddings for biomedical NLP. In: Proceedings of the

15th workshop on biomedical natural language processing. https://doi.org/10.18653/v1/W16-2922
31. Li Y, Tarlow D, Brockschmidt M, Zemel R (2016) Gated graph sequence neural networks. In: Proceedings of the 4th international conference on learning representations
32. Song L, Zhang Y, Wang Z, Gildea D (2018) A graph-to-sequence model for amr-to-text generation. In: Proceedings of the 56th annual meeting of the association for computational linguistics. https://doi.org/10.18653/v1/P18-1150
33. Zhang Y, Liu Q, Song L (2018) Sentence-state LSTM for text representation. In: Proceedings of the 56th annual meeting of the association for computational linguistics. https://doi.org/10.18653/v1/P18-1030
34. Jia R, Wong C, Poon H (2019) 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. https://doi.org/10.18653/v1/n19-1370
35. Yao Y, Ye D, Li P, Han X, Lin Y, Liu Z, Liu Z, Huang L, Zhou J, Sun M (2019) DocRED: A large-scale document-level relation extraction dataset. In: Proceedings of the 57th annual meeting of the association for computational linguistics. https://doi.org/10.18653/v1/p19-1074
36. Li J, Sun Y, Johnson RJ, Sciaky D, Wei C-H, Leaman R, Davis AP, Mattingly CJ, Wiegers TC, Lu Z (2016) Biocreative v cdr task corpus: a resource for chemical disease relation extraction. Database J Biol Databases Curation 2016. https://doi.org/10.1093/database/baw068
37. Wu Y, Luo R, Leung HC, Ting H-F, Lam T-W (2019) Renet: A deep learning approach for extracting gene-disease associations from literature. In: Proceedings of the 23rd international conference on research in computational molecular biology. https://doi.org/10.1007/978-3-030-17083-7_17
38. Tan Q, He R, Bing L, Ng HT (2022) Document-level relation extraction with adaptive focal loss and knowledge distillation. In: Findings of the association for computational linguistics: ACL 2022. https://doi.org/10.18653/v1/2022.findings-acl.132
39. Paszke A, Gross S, Chintala S, Chanan G, Yang E, DeVito Z, Lin Z, Desmaison A, Antiga L, Lerer A (2017) Automatic differentiation in pytorch. In: Proceedings of NIPS 2017 workshop
40. Kingma DP, Ba J(2015) Adam: a method for stochastic optimization. In: Proceedings of the 3th international conference on learning representations
41. Caruana R, Lawrence S, Giles CL (2000) Overfitting in neural nets: backpropagation, conjugate gradient, and early stopping. In: Proceedings of the advances in neural information processing systems 13
42. Devlin J, Chang M-W, Lee K, Toutanova K (2019) 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: human language technologies. https://doi.org/10.18653/v1/n19-1423
43. Liu Y, Ott M, Goyal N, Du J, Joshi M, Chen D, Levy O, Lewis M, Zettlemoyer L, Stoyanov V (2019) Roberta: A robustly optimized bert pretraining approach. arXiv:1907.11692
44. Beltagy I, Lo K, Cohan A (2019) SciBERT: A pretrained language model for scientific text. In: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing . https://doi.org/10.18653/v1/D19-1371
45. Huang K, Wang G, Ma T, Huang J (2020) Entity and evidence guided relation extraction for docred. arXiv:2008.12283
46. Xie Y, Shen J, Li S, Mao Y, Han J (2021) Eider: Evidence-enhanced document-level relation extraction. arXiv:2106.08657
47. Li B, Ye W, Sheng Z, Xie R, Xi X, Zhang S (2020) Graph enhanced dual attention network for document-level relation extraction. In: Proceedings of the 28th international conference on computational linguistics. https://doi.org/10.18653/v1/2020.coling-main.136



Tingting Hang received her Ph.D. degree from Hohai University, Nanjing, China, in 2022. Since 2022, she has been with Anhui University of Technology, China, where she is currently a Lecturer. Her current research interests include information extraction, and domain knowledge graph construction.

Jun Feng received her Ph.D. degree from Nagoya University, Nagoya, Japan, in 2004. Since 1994, she has been with Hohai University, China, where she is currently a Professor. Her research interests include data management, domain knowledge discovery research, and water conservancy informatization.

Yunfeng Wang received his M.S. degree from Hohai University, Nanjing, China, in 2022. His research interests include causal structure learning, hydro-logical time series data mining, and deep learning modeling.

Le Yan received the Ph.D. degree from Hohai University, Nanjing, China, in 2021. Since 2021, he has been with Nanjing Xiaozhuang University, China, where he is currently a Lecturer. His research interests include hydro-logical spatial-temporal data mining, hydro-logical time series data prediction, and deep learning modeling.