Inter-sentence Entity Relation Extraction based on GNN of Message Propagation

Chen Qian
Shanxi University

Guan Chunxiang
Shanxi University

Xin Guo (guoxinjsj@sxu.edu.cn)
Shanxi University https://orcid.org/0000-0002-8627-1199

Wang Suge
Shanxi University

Li Deyu
Shanxi University

Research Article

Keywords: Inter-sentence Entity Relation Extraction, Relational Graph Convolutional Neural Network, Message Propagation

Posted Date: February 15th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1202426/v1

Inter-sentence Entity Relation Extraction based on GNN of Message Propagation

Chen Qian[1,2], Guan Chunxiang[1], Guo Xin[1*], Wang Suge[1,2] and Li Deyu[1,2]

[1*]School of Computer and Information Technology, Shanxi University, Wucheng Road, Taiyuan, 030006, Shanxi, China.

[2]Key Laboratory Computational Intelligence and Chinese Information Processing of Ministry of Education, Shanxi University, Wucheng Road, Taiyuan, 030006, Shanxi, China.

*Corresponding author(s). E-mail(s): guoxinjsj@sxu.edu.cn;
Contributing authors: chenqian@sxu.edu.cn; 2640551793@qq.com; wsg@sxu.edu.cn; lidy@sxu.edu.cn;

Abstract

Compared with sentence-level relation extraction, it is more common that entities involving relationships exist in multiple sentences in actual scenarios. Therefore, document-level relation extraction has gradually become a research hotspot in the field of information extraction in recent years. In order to make full use of the contextual information and structure information of the document, this paper combine embedding representation of entities and the explicit structural relationships between entities to study extraction of document-level inter-sentence relations. Firstly, document is encoded using BiLSTM, while document graph is established using the explicit structure of document; Secondly, relational graph convolutional neural network based on message propagation is used to update nodes in the graph, thus local information of document is integrated into node embedding representation, using which edge embedding is updated; Finally, an iterative algorithm is used to complete the reasoning of edge information, and entity relation is predicted using a feed-forward network based on edge embedding. Experimental results show that, compared with the EoG model, the F1 values of inter-sentence relation are increased by 2.6% and 0.8% on the CDR and GDA datasets, respectively.

Keywords: Inter-sentence Entity Relation Extraction, Relational Graph Convolutional Neural Network, Message Propagation

1 Introduction

Relation Extraction (RE) is one subtask of information extraction. The goal is to extract the relation between entity pairs in plain text. It is a fundamental research direction for downstream applications such as knowledge graph construction, question answering systems, et al.[12]. Traditional relation extraction only focuses on sentence and entity pairs exist in the same sentence(e.g Jose L et al.[15] extract information from sentences to create RDF triples about relationship facts.) However, in the fields of medicine, finance, and tourism, most relationship facts between entities are obtained through cross-sentence. An analysis result of the Wikipedia Corpus shows that at least 40% relationship facts are obtained through document-level relation extraction[22]. Therefore, relation extraction from sentence-level

to document-level has important practical application value. The extraction of document-level inter-sentence relation mainly faces two challenges:

(1) There may be multiple mentions of an entity in the document. TACRED[1] and SemEval-2010 Task 8[10] and other sentence-level relation extraction datasets contain only one entity pair in each sentence, and each entity does not have multiple mentions. However, the document generally contains multiple entity pairs and an entity may have different mentions with a high probability. We need to pay attention to different mentions of an entity, and finally get all possible relationship fact triples.

(2) The document generally includes multiple relationship fact triples. There may be a logical relation between them. Therefore, the model is required to have a certain ability of document reasoning.

A specific example of document-level inter-sentence relation extraction is shown in Figure 1. The shaded words in bold represent entity mention. In order to predict inter-sentence relation between angiotensin-converting enzyme inhibitor (ACE) and urticaria, two different mentions of ACE inhibitors in the first and third sentence of the document should be recognized first; it should be concluded that adverse drug reactions are the cause of urticaria and angioedema from the first sentence "Adverse reactions to drugs are well recognized as a cause of acute or chronic urticaria, and angio-oedema. "; Moreover, one should conclude that the use of ACE inhibitors can cause acute angioedema according to the third sentence "Soon after the introduction of ACE inhibitors, acute bouts of angio-oedema were reported in association with the use of these drugs.". In summary, it can be inferred that ACE inhibitors are the cause of urticaria. Therefore, for inter-sentence relation extraction at the document-level, the model needs to make logical inference on multiple sentences based on the understanding of the document and then predict the relation between entity pairs. It is obviously beyond the scope of sentence-level relation extraction model.

For document-level inter-sentence relation extraction, EoG model proposed by Fenia Christopoulou et al.[6] used the embedding representation of edges which fuse relation information to predict relation. Shuang Zeng et al.[23] proposed to fuse the relation information into node embedding using double graphs. Damai Dai et al.[7] performed entity relation prediction by using entity embedding representation with different granularities. Above methods have achieved good results on the task of document-level inter-sentence relation extraction, but these models either only use node embedding representation that integrate local document information, or only use edge embedding representation that integrate relation information. We propose a document-level inter-sentence relation extraction model based on message propagation graph neural network, which is able to fully combine node embedding representation and edge embedding representation, and integrate context information and document structure information.

The main contributions of this paper are as follows: (1) A document-level inter-sentence relation extraction model based on message propagation graph neural network is proposed. This model uses message propagation graph neural network to update the graph nodes, and obtains node embedding representation that integrates local document information to update edge embedding representation. (2) Compared with the EoG model, our model increase F1 of inter-sentence relation extraction by 2.6% and 0.8% on the CDR and GDA datasets respectively.

2 Related work

In order to identify document-level inter-sentence relation between entity pairs, the complex interactions of multiple entities need to be modeled. Document-level inter-sentence relation extraction models can be broadly classified into two categories: sequence-based approaches and graph-based approaches. Sequence-based approaches mainly model document structure information by using Recurrent Neural Networks(RNNs), in which entity embedding representation can be obtained. Transformer can implicitly model long-distance dependencies[18] and Transformer-based pre-training model has been widely used for document structure modeling in recent years. Chaojun Xiao et al.[20]used Bert to encode document structure information for Document-level inter-sentence relation extraction; Benfeng Xu et al.[21] proposed SSAN model based on Transformer, which incorporated graph information into the

[1] Adverse reactions to drugs are well recognized as a cause of acute or chronic **urticaria**, and **angio-oedema**. [2] **Angiotensin-converting enzyme (ACE) inhibitors**, used to treat **hypertension** and **congestive heart failure**, were introduced in Europe in the middle of the eighties, and the use of these drugs has increased progressively. [3] Soon after the introduction of **ACE inhibitors**, acute bouts of **angio-oedema** were reported in association with the use of these drugs.[4] We wish to draw attention to the possibility of adverse reactions to **ACE inhibitors** after long-term use and in patients with pre-existing **angio-oedema**.

Fig. 1 An example from CDR dataset.

encoder's self-attention; ZhouW et al.[25] proposed a document-level relation extraction model based on BERT with adaptive thresholding and local contextual pooling; Eberts et al.[8] combine the entity representation at the global level with the mention representation at the local level in the relation classification task.

In recent years, graph-based approaches have gradually become the mainstream technique for document-level relation extraction due to the advantages of graph neural networks in mining document structure features. Zhenyu Zhang et al.[24] proposed a two-layer heterogeneous graph that separates document structure modeling layer from relation inference layer; Guoshun Nan et al.[16] proposed a LSP model, which used syntactic dependencies to construct graph. Graph nodes automatically proceed to learn more non-neighbor information through neighbor nodes in a fully connected state. Graph Neural Networks(GNN) is a natural extension of traditional sequence-based neural network, and it is widely used to model complex graph structure data especially document embedding representation. GNN can handle long distance dependency problem easily, so we adopt GNN in our document-level relation extraction model. Although the current work has made great progress, existing works either only considered the embedding representation of nodes or the embedding representation of edges, lacking the combination of both leading to not good F1 in this task. The major challenges of current graph-based model is how to construct nodes and edges in the graph and how to perform document information propagation and inference on the constructed graph. In this paper, we propose to incorporate Message Propagation mechanisims in GNN by considering both nodes and edges embedding representation for document-level inter-sentence relation extraction, and experiment results show that our model exceed baseline EoG model.

3 Inter-sentence Entity Relation Extraction model based on GNN of Message Propagation

We propose a document-level inter-sentence relation extraction model based on message propagation graph neural network. Figure 2 shows the overall architecture of model. Different from EoG model, our model uses message propagation graph neural network to update embedding representation of nodes in which local document information can be fused in node embedding. Then edge embedding is updated using the latest node embedding representation once again to finally make the prediction of the relation. The model is divided into four parts: (1) Encoding layer: The embedding layer converts document information into initial text vector representation. In order to obtain representation of word context information, we use BiLSTM encoder; (2) graph construction layer: Word context representation is used to construct graph structure according to pre-defined node types and edge types; (3) Inference layer: Message propagation in GNN is used to update node representation by fusing document local information. Finally we make inference over edges by an iterative algorithm; (4) Classification layer: Edge embedding representation between entity pairs is used to predict the type of relation. For brevity, Figure 2 omits some edges in the graph construction layer. e_1 denotes entity that corresponding mention appeared in the first time in the document, and $m_2^{[1]}$ denotes the mention that correspond to the second entity that

appears in the first sentence, which stands for *cacaine* in Figure 2, and s_1 denotes the first sentence. Before specifying the details of model, we first introduce task definition.

3.1 Task definition

Given a labeled document $D = \{s_i\}_{i=1}^{N}$ and $s_i = \{w_j\}_{j=1}^{M}$, s_i denotes the i-th sentence in the document D, and w_j denotes the j-th word in sentence s_i, in which N and M stands for the number of sentences in the document and words in the i-th sentence respectively. The entity set $\varepsilon = \{e_p\}_{p=1}^{P}$ contains P entities, and the p-th entity $e_p = \{m_q\}_{q=1}^{Q}$ contains Q mentions. The task of this paper is to predict relation through the probability of relation between all entities pairs predicted by model in a given document. Document D, entity set ε and pre-defined relation set R is the input of the model, and relation fact triple set $\{(s, r, o) \| s, o \in \varepsilon, r \in R\}$ can be output.

3.2 Encoding layer

In embedding layer, input document is initialized as vector representation by word embedding, and then BiLSTM [11] is used to encode the document to get context embedding. Finally, context vector representation of each word is obtained. Suppose initial text vector is represented as $\boldsymbol{D} = \{\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_i, \cdots\}$, where $\boldsymbol{w}_i$ is initial vector representation of the i-th word. After going through BiLSTM encoder, the context vector is actually the hidden state representation of the i-th word, which is shown in formula (1), where h is the dimension of hidden state after the concatenation of the forward and the backward LSTM unit.

$$\boldsymbol{h}_i = \left[\overrightarrow{LSTM}(\boldsymbol{w}_i), \overleftarrow{LSTM}(\boldsymbol{w}_i)\right] \in \mathbb{R}^h \quad (1)$$

3.3 Graph construction layer

In this section, the output of encoder is used to construct graph structure, and different types of nodes and edges are defined to capture different dependency information of document. We use three types of nodes, i. e. sentence nodes, mention nodes, and entity nodes. Sentence node representation can be obtained by mean-pooling over all words vector in the sentence. Mention node representation can be obtained by similar mean-pooling over all words vector contained by this mention. Entity node representation can be calculated by the same mean-pooling over all mention nodes corresponding to this entity. In order to distinguish different node types in the constructed graph, the final node representation is the concatenation of node embedding and corresponding node type embedding. The embedding represention of these three node types can be formulated by (2)-(4), where $\boldsymbol{t}_s$, $\boldsymbol{t}_e$, $\boldsymbol{t}_m \in \mathbb{R}^t$ are embeddings of node type for sentence, entity and mention respectively, $\boldsymbol{n}_{s_i}$, $\boldsymbol{n}_{e_p}$, $\boldsymbol{n}_{m_q} \in \mathbb{R}^{t+h}$ are node embeddings for sentence, entity and mention respectively, t indicates the dimension of node type embedding.

$$\boldsymbol{n}_{s_i} = [\frac{1}{M}\sum_{j=1}^{M}\boldsymbol{h}_j, \boldsymbol{t}_s] \quad (2)$$

$$\boldsymbol{n}_{e_p} = [\frac{1}{K-j+1}\sum_{j}^{K}\boldsymbol{n}_{m_q}, \boldsymbol{t}_e] \quad (3)$$

$$\boldsymbol{n}_{m_q} = [\frac{1}{Q}\sum_{j=1}^{Q}\boldsymbol{h}_j, \boldsymbol{t}_m] \quad (4)$$

Based on three types of nodes, five types of undirected edges listed below are constructed to represent different interaction between various nodes. The calculation of vector embedding of edge is the same as EoG model[6].

1. Mention-Mention Edge (MM): An MM edge is established between two mentions both of which appeared in the same sentence.
2. Mention-Entity Edge (ME): An ME edge is constructed between mention m_q and corresponding entity e_p.
3. Mention-Sentence Edge (MS): An MS edge is constructed between mention m_q and sentence s_i that m_q belongs to.
4. Entity-Sentence Edge (ES): If one of the mentions corresponding to e_p is in the sentence s_i, ES edge are constructed between e_p and s_i.
5. Sentence-Sentence Edge (SS): SS edges are constructed by fully connecting all sentences in the document.

In order to facilitate subsequent update and inference, a linear transformation is used to unify the dimension of edges embedding, as shown in formula (5).

$$\tilde{\boldsymbol{u}}_{\boldsymbol{o}} = \boldsymbol{W}_o \boldsymbol{u}_o, \quad (5)$$

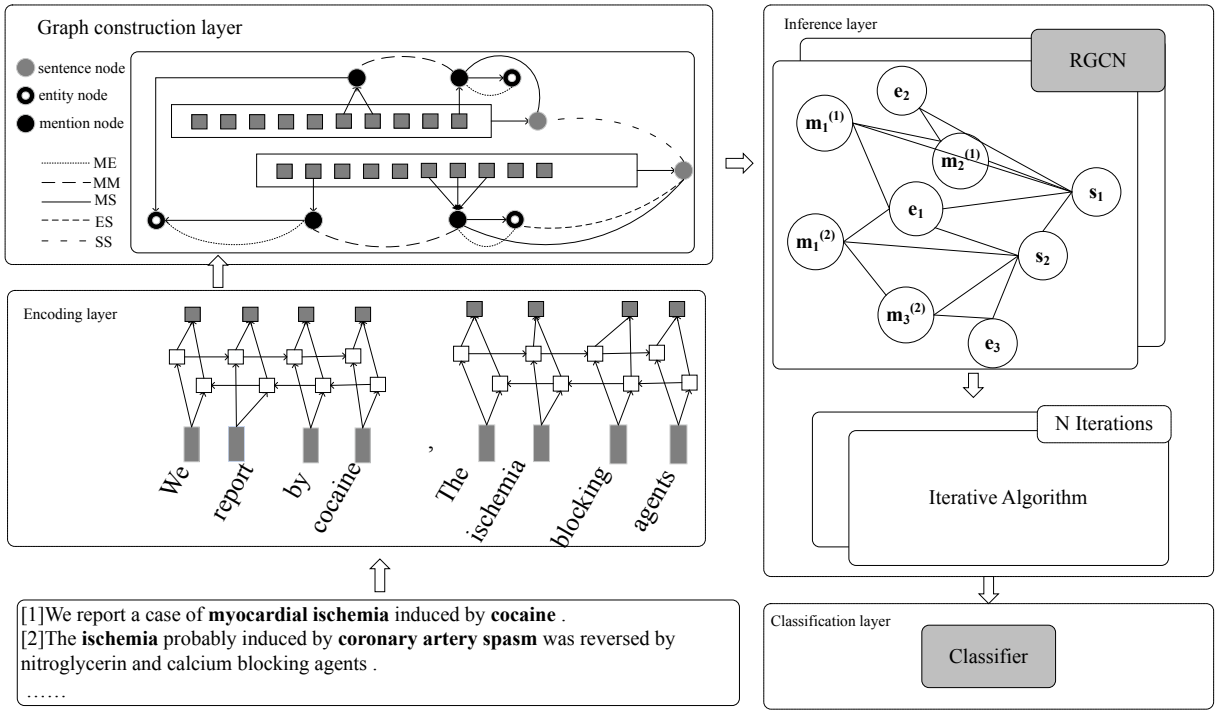

Fig. 2 The overall framework of our model

where $\boldsymbol{u}_o \in \mathbb{R}^{d_e}$ represents all edges before the dimension is unified, $\tilde{\boldsymbol{u}}_{\boldsymbol{o}} \in \mathbb{R}^d$ represents all edges after the dimension is unified, $\boldsymbol{W}_o \in \mathbb{R}^{d_e \times d}$, $o \in O = \{MM, ME, MS, ES, SS\}$.

3.4 RGCN Message Propagation with both Edge and Node Embeddings

In order to integrate local information of the document, each node of the graph constructed in the previous section needs to be updated by message propagation mechanism. There are many ways to update nodes in GNN. In this paper, we use Relational Graph Convolutional Networks(RGCN) to update nodes[17]. Unlike graph convolutional neural network(GCN), RGCN updates all graph nodes at the same time, and furthermore graph information can be modeled with multiple relations. When message is propagated in the graph, RGCN integrates local information of the document by aggregating neighbor nodes of different edge types. If the i-th node of the l-th layer is expressed as $\boldsymbol{n}_i^l$, the update result of the $l+1$-th layer of the i-th node can be calculated by

$$\boldsymbol{n}_i^{l+1} = \sigma(\sum_{o \in O} \sum_{j \in U_i^o} \boldsymbol{W}_o^l \boldsymbol{n}_j^l + \boldsymbol{W}_0^l \boldsymbol{n}_i^l), \quad (6)$$

where $\boldsymbol{W}_o^l$, $\boldsymbol{W}_0^l \in \mathbb{R}^{d \times d}$, and σ represent sigmoid function. U_i^o represents set of neighbor nodes connected to the i-th node whose edge type is o.

The embedding of all nodes in the graph are updated by fusing local document information with RGCN, and finally the initial embedding of edges are updated using the latest node embedding.

3.5 Inference Layer

After updating node and edge information, we use an iterative algorithm to generate embeddings of the edge between each entity pair[5]. The algorithm is specifically divided into two steps in each iteration.

The first step is path generation. Suppose a-th node is an intermediate node between the i-th and the j-th node, a path representation between these two node is generated. Bilinear transformation is performed to generate a path representation for edges $\boldsymbol{u}_{ia}$ and $\boldsymbol{u}_{aj}$ connecting the i-th and j-th

node to the the a-th node which can be shown in formula (7).

$$\boldsymbol{p}_{iaj}^{(\lambda)} = \sigma(\boldsymbol{u}_{ia}^{(\lambda)} \odot (\boldsymbol{W}_g \boldsymbol{u}_{aj}^{(\lambda)})), \tag{7}$$

where $W_g \in \mathbb{R}^{d\times d}$ is a trainable parameter. $\odot$ denotes element-wise multiplication, and λ is the path length between the two nodes.

The second step is path aggregation. Multiple path representations between the i-th and the j-th node is aggregated to obtain unique path representation, as shown in formula (8), where β is used to control the aggregation degree of multiple paths between the i-th and the j-th node, and is a pre-defined hyperparameter.

$$\boldsymbol{u}_{ij}^{(2\lambda)} = \beta \boldsymbol{u}_{ij}^{(\lambda)} + (1-\beta) \sum_{a\neq i,j} \boldsymbol{p}_{iaj}^{(\lambda)} \tag{8}$$

After edge reasoning is completed, the embedding of edge $\boldsymbol{u}_{EE}$ is obtained between each entity pair, which can be used for relation prediction.

3.6 Classification Layer

In the classification layer, the embedding of edge $\boldsymbol{u}_{EE}$ for each entity pair is used to predict relation by a single FFN with softmax. The classifier can be fomulated in formula (9), where $\boldsymbol{W}_c \in \mathbb{R}^{d\times R}$,$\boldsymbol{b}_c \in \mathbb{R}^R$. In training stage, cross-entropy loss function is used as the objective function. Formula (10) gives the loss calculation of one sample, where $\boldsymbol{y}_r = [y_1, y_2, \cdots, y_r, \cdots] \in \mathbb{R}^R$ represents relation type label predicted by our model, y_r^* represents ground-truth relation type label.

$$\boldsymbol{y} = softmax(\boldsymbol{W}_c \boldsymbol{u}_{EE} + \boldsymbol{b}_c) \tag{9}$$

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

4 Experimental results and analysis

4.1 Datasets

To our knowledge, there are few public datasets involving document-level relation extraction. Existed datasets mainly focus on medical domain and open domain. We choose two extensive document-level relation extraction datasets in the biomedical field, i.e. CDR (Chemical Disease Relation) dataset[1] and GDA (Gene Disease Associations) dataset[2]. Since there is large difference in the number of relationship facts triple between these two datasets, the stability of our model can be verified based on CDR and GDA. Different from the extraction of protein–protein interactions (PPIs)[4], we study Chemical-Disease interaction relations in CDR and Gene-Disease interaction relations in GDA.

CDR dataset is a chemical-disease interaction dataset constructed in 2016[14], which annotates the binary relation between chemistry and disease concept. It is composed of 1500 abstracts, which are divided into three equal-sized set for training, test and development. Both start and end position of entity mention in each document was annotated in CDR, in which two relation types are predefined between chemical-disease entity pairs. Besides, each entity is assigned with a concept identifier from the biomedical vocabulary. The detailed statistics for CDR are shown in Table 1.

GDA is a genes-disease relation extraction dataset released in 2019[19], which annotates the binary relation between gene and disease concept. It contains 30,192 abstracts, in which 29,192 abstracts are used for training and 1,000 are used as test set. Development set is obtained from 20% of training set. The annotation content in GDA is the same as that in CDR, while GDA annotates relations between gene-disease entity pairs. The detailed statistics for GDA are shown in Table 2.

4.2 Experimental setup

All experiments are carried out under the framework of Pytorch. Due to the small size of training set in CDR, we use both training set and development set to train the model. PubMed word embedding[3] is used to initialize token embedding in CDR, and the learning rate is set to 0.002; Random word embedding is used to initialize token embeddings in GDA, and the learning rate is set to 0.0001. For other parameter, the dimensions of word embedding, the BiLSTM unit, the node type embedding, and the node embedding in RGCN are set to 200, 100, 10 and 210 respectively for both

[1] CDR dataset: https://biocreative.bioinformatics.udel.edu/resources/corpora-v-cdr-corpus

[2] GDA dataset: https://www.disgenet.org/downloads.

Table 1 Statistics of CDR dataset.

dataset	document	chemistry mention	chemistry entity	disease mention	disease entity	relation	intra-sentence	Inter-sentence
train	500	5203	1467	4182	1965	1038	754	284
test	500	5385	1435	4424	1988	1066	747	319
dev	500	5347	1507	4244	1865	1012	765	246

Table 2 Statistics of GDA dataset.

dataset	document	gene mention	gene entity	disease mention	disease entity	relation	intra-sentence	Inter-sentence
train	29192	205457	46151	226015	67257	36079	30199	5880
test	1000	8404	1903	9524	2778	1502	1273	229
dev	5839	51410	11406	56318	16703	8762	7408	1354

CDR and GDA. Adam[13] is used to optimize the model. In order to verify the effectiveness of proposed model, we compare it with the following baseline models:

(1) ME-CNN[9]: ME-CNN model uses the maximum entropy and the convolutional neural network to extract document-level inter-sentence relation.

(2) APG with Dep.Graphs[2]: The model uses a supervised kernel method.

(3) BERTbase+SSANdecomp[21]: This model is a sequence-based relation extraction model. The graph structure information is added to the self-attention layer of a Transformer encoder to obtain the final entity embedding, which is used to predict relation between each entity pair.

(4) EoG[6]: EoG model construct a graph to model each document, and after edge reasoning, relation between each entity is predicted over corresponding edge in the graph.

(5) LSR[16]: LSR model generate graph over each document, and adjusts graph structure dynamically for reasoning over edges and finally relation is classified.

(6) LSR w/o MDP Nodes[16]: Unlike LSR model, this model removes the MDP node type and constructs a fully connected graph at the same time.

Among them, (1)-(3) are sequence-based models, and (4)-(6) are graph-based models. We use three evaluation metrics, i.e. the intra-sentence F1(Intra F1), the inter-sentence F1(Inter F1) and the overall F1(overall F1) which consider both inter-sentence and intra-sentence entity relation, respectively.

4.3 Experimental results

Experiment results of CDR dataset are shown in Table 3. Compared with the sequence-based models, our model has improved in terms of all three metrics. Among them, the inter-sentence F1 increased by 41.8%, 7.8%, and 8.4%, respectively%. For graph-based models, the inter-sentence F1 increased by 2.6%, 0.4%, and 3.2%, respectively. Obviously, the improvement of graph-based models are not as high as that of sequence-based models, which indicate that graph-based models can better model document than sequence-based models. This may also be the reason why graph-based methods are popular in document-level relation extraction in recent years.

In order to further demonstrate stability of the model in document-level inter-sentence relation extraction, experiments were carried out on GDA dataset which is much larger than CDR dataset in scale. Experimental results over GDA are shown in Table 4, from which we can see that inter-sentence F1 increases by 0.8% and 1.2% compared with EoG and LSR. However it is lower than the experimental results when the model is changed to a fully connected graph. It may be due to the inter-sentence relationship fact triples account for 16% of total number of relationship fact triples in GDA dataset, while inter-sentence relationship fact triples account for 37% in CDR dataset. The small proportion of inter-sentence relationship fact triples makes the model fail to learn document structure information and context information well.

It can be included from Table 3, and Table 4 that our model can enhance the ability to extract inter-sentence relation by fusing node information

into edge embedding through RGCN. Besides, our model perform well over both CDR and GDA datasets although it is quite different between these two datases in scale, which verify the stability of our model over different scaled dataset.

4.4 Experimental analysis

In order to further verify the effectiveness of the document-level inter-sentence relation extraction model based on GNN of Message Propagation, the CDR dataset with more inter-sentence relationship fact triples are selected for ablation experiments.

Table 5 shows the experimental results of different word embedding. When PubMed word embedding is adopted, the overall F1 value, intra-sentence F1 value and inter-sentence F1 value are 7.1%, 5.7% and 10.5% higher than random initial word embedding, respectively. Compared with the traditional Glove word embedding, these three metrics are also increased by 6%, 5.9%, 6.8%, respectively. We can see that model using word embedding corresponding to the target field can outperform model using open-domain word embedding and random embedding. In order to select the best number of RGCN layer, we compared models with different layers of RGCN and results are shown in Figure 3. It shows that overall F1, intra-sentence F1, and inter-sentence F1 reached the highest values, which are 72.4%, 67%, and 53.5%, respectively if the number of RGCN layers set to 2. It can be seen from the Figure 3 that as the number of layers increases, all F1 starts to increase steadily. It may be because the model continuously aggregates the local information of document to make graph node information more abundant. However, when the number of RGCN layers exceeds two layers, the F1 gradually decreases, which may cause the graph nodes to aggregate many redundant and irrelevant information in the document. So we choose 2 as the number of RGCN layers.

We also conduct experiment with different inference times over CDR test set, in order to select the best inference times, and results are shown in Figure 4. It can be seen that if the number of inference time is 4, overall F1, intra-sentence F1 and inter-sentence F1 are 67%, 72.4% and

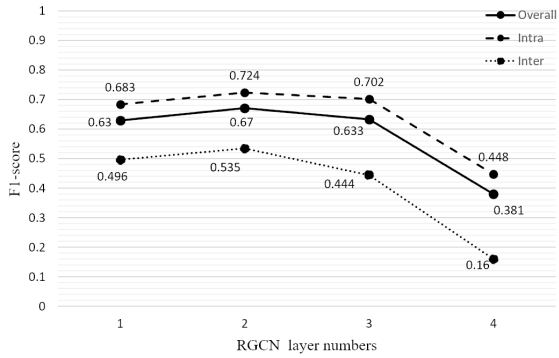

Fig. 3 Experimental results of RGCN with different layers

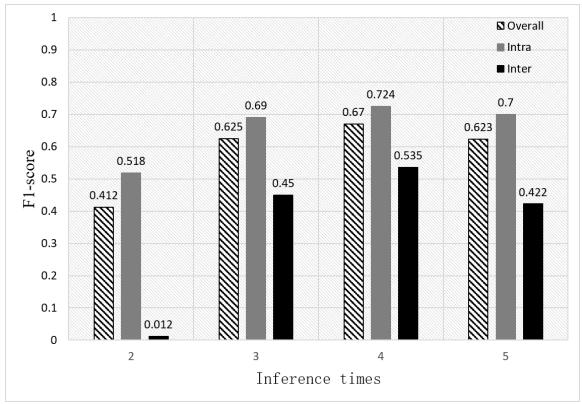

Fig. 4 Experimental results with inference different times

53.5% respectively, which is the best result. However, if the number of inference time is too small, the model may lose some logical reasoning information; Reversely, too many inference times will make edge embedding contain a lot of redundant information leading lower performance in logical reasoning. In order to verify the importance of each edge information defined in the graph, we removed each type of edges from the graph once a time and conducted 5 experiments separately, and results are shown in Table 6, in which overall F1, intra-sentence F1, and inter-sentence F1 decreased by 11.3%, 5.2%, and 49.9% when SS edges are removed. This indicate that interactive information between sentences has an important influence on the recognition of intra-sentence and inter-sentence relational triples. In addition, it can be concluded from Table 6 that no matter which type of edge is removed, it will have poor impact over the performance of inter-sentence relation extraction. So all five edge type should be used in the model, especially SS. The removal of edge MS

Table 3 Experimental results of CDR test set.

Model	Overall F1(%)	Intra F1(%)	Inter F1(%)
ME-CNN(2017)	61.3	57.2	11.7
APG with Dep.Graph(2018)	60.3	65.1	45.7
EoG(2019)	63.6	68.2	50.9
BERTbase+SSANdecomp(2020)	61.2	68.6	45.1
LSR(2020)	61.2	66.2	50.3
LSR w/o MDP Nodes(2020)	64.8	68.9	53.1
ours	**67.0**	**72.4**	**53.5**
ours-FULL	62.1	69.5	42.9

Table 4 Experimental results of GDA test set.

Model	Overall F1(%)	Intra F1(%)	Inter F1(%)
EoG(2019)	**81.5**	**85.2**	50.0
LSR(2020)	79.6	83.1	49.6
ours	80.7	85.1	50.8
ours-FULL	81.0	85.0	**54.3**

and ES reduces the performance of model in terms of Inter-F1 imply that sentence node is the most important node type as it can propagate relation information accoss sentence through sentence node. Exclude sentence node, we observe that the absence of mention node seems bring worse impact than the absence of entity node in the inter-sentece scenario while in intra-sentence scenario it is not the case. This indicate that mention node is more important than entity node cross sentence while entity node is more important than mention node inside a sentence for relation extraction. This may due to the fact that fewer mention for a target entity exist inside a sentence, while multiple mention involving the same entity scattered acorss multiple sentences.

Table 7 shows an example of error prediction by our model in CDR test set. In this example, there is a inter-sentence relation between two entities "corticosteroid" and "systemic sclerosis". The golden label is a CID relation which indicate the chemistry in the subject induce a certain disease in the object, but the prediction result is an NR relation which indicate the subject have nothing to do with the object. In the relationship facts of the CDR test set, the number of CID relation is 319, the number of NR relation is 2593. The number of various relations are imbalanced. The phenomenon may be one of the main reasons why some inter-sentence relationship facts are predicted incorrectly.

5 Conclusion

The paper proposes a document-level inter-sentence entity relation extraction based on GNN of message propagation. On the basis of EoG model, RGCN is added to update the information of graph node so as to fuse local document information, then the model uses the embedding information of graph node to update the edge information, and the whole message propagation mechanism is the foundation of subsequent reasoning and classification. Experimental results show that, compared with the sequence-based models and existed graph-based models, our model has improved the performance of inter-sentence relation extraction over CDR and GDA in terms of inter-F1. However, we notice that two datasets have an imbalance distribution of intra-sentence and inter-sentence relationship fact triples. This type of data imbalance problem pose great challenge in inter-sentence relation extraction. Although the proportion of inter-sentence relationship fact triples in the document is small, they should not be ignored in practical applications.

In the future, our research may focus on how to accurately extract inter-sentence relation with a small number of distributed inter-sentence relationship fact triples. The number of each relation type in both datasets has obvious long-tail distribution in the document-level relation extraction. This imbalanced distribution of relation types can

Table 5 Experimental results of different word embedding for CDR test set.

Word embedding type	Overall F1(%)	Intra F1(%)	Inter F1(%)
PubMed	**67.0**	**72.4**	**53.5**
Random	59.9	66.7	43.0
Glove	61.0	66.5	46.7

Table 6 Experimental results with different edges removed from graph

model	Overall F1(%)	Intra F1(%)	Inter F1(%)
ours	67.0	72.4	53.5
-MM	63.3	70.1	45.0
-ME	62.8	69.1	45.7
-MS	62.2	69.3	43.8
-ES	61.9	68.2	45.4
-SS	55.7	67.2	3.6

greatly affect the performance of extraction in the document, which is also a problem that needs to be solved urgently in the future.

Acknowledgments. This work was supported by the National Science Foundation of Shanxi Province under Grant No.201901D111032, the Key Research and Development project of Shanxi Province under Grant No. 201803D421024, and Enterprise Cooperation Project of Xiaodian district in Shanxi Province No. 2020XDCXY05.

Declarations

Some journals require declarations to be submitted in a standardised format. Please check the Instructions for Authors of the journal to which you are submitting to see if you need to complete this section. If yes, your manuscript must contain the following sections under the heading 'Declarations':

- The data that support the findings of this study is publicly available for noncommercial use.
- All authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest or non-financial interest in the subject matter or materials discussed in this manuscript.

References

[1] Christoph Alt, Aleksandra Gabryszak, and Leonhard Hennig. TACRED revisited: A thorough evaluation of the TACRED relation extraction task. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1558–1569, Online, jul 2020. Association for Computational Linguistics.

[2] Nagesh Chandrasekarasastry, Karin Verspoor, Trevor Cohn, and Kotagiri Ramamohanarao. Exploiting graph kernels for high performance biomedical relation extraction. *Journal of Biomedical Semantics*, 9, 01 2018.

[3] Billy Chiu, Gamal Crichton, Anna Korhonen, and Sampo Pyysalo. How to train good word embeddings for biomedical NLP. In *Proceedings of the 15th Workshop on Biomedical Natural Language Processing*, pages 166–174, Berlin, Germany, aug 2016. Association for Computational Linguistics.

[4] Sung-Pil Choi. Extraction of protein–protein interactions (ppis) from the literature by deep convolutional neural networks with various feature embeddings. *Journal of Information Science*, 44(1):60–73, 2018.

[5] Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. A walk-based model on entity graphs for relation extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 81–88, Melbourne, Australia, jul 2018. Association for Computational Linguistics.

[6] Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 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)*, pages 4925–4936, Hong Kong, China, nov

Table 7 Error prediction results of CDR test set

[1] Late - onset scleroderma renal crisis induced by tacrolimus and prednisolone : a case report .
[2] Scleroderma renal crisis (SRC) is a rare complication of systemic sclerosis (SSc), but can be severe enough to require temporary or permanent renal replacement therapy .
[3] Moderate to high dose corticosteroid use is recognized as a major risk factor for SRC .
[4] Furthermore , there have been reports of thrombotic microangiopathy precipitated by cyclosporine in patients with SSc .
[5]In this article , we report a patient with SRC induced by tacrolimus and corticosteroids .
[6]The aim of this work is to call attention to the risk of tacrolimus use in patients with SSc .
Prediction: 1:NR:2
Truth: 1:CID:2
Type: CROSS
Arg1: D000305 ‖ corticosteroid ‖ corticosteroids
Arg2: D012595 ‖ systemic sclerosis ‖ SSc ‖ SSc ‖ SSc

2019. Association for Computational Linguistics.

[7] Damai Dai, Jing Ren, Shuang Zeng, Baobao Chang, and Zhifang Sui. Coarse-to-Fine Entity Representations for Document-level Relation Extraction. *arXiv e-prints*, page arXiv:2012.02507, dec 2020.

[8] Markus Eberts and Adrian Ulges. An end-to-end model for entity-level relation extraction using multi-instance learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3650–3660, Online, apr 2021. Association for Computational Linguistics.

[9] Jinghang Gu, Fuqing Sun, Longhua Qian, and Guodong Zhou. Chemical-induced disease relation extraction via convolutional neural network. *Database*, 2017, 04 2017. bax024.

[10] Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. SemEval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 33–38, Uppsala, Sweden, jul 2010. Association for Computational Linguistics.

[11] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 11 1997.

[12] Haiyun Jiang, Qiaoben Bao, Qiao Cheng, Deqing Yang, Li Wang, and Yanghua Xiao. Complex relation extraction: Challenges and opportunities, 2020.

[13] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014. cite arxiv:1412.6980Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015.

[14] Jiao Li, Yueping Sun, Robin J. Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J. Mattingly, Thomas C. Wiegers, and Zhiyong Lu. BioCreative V CDR task corpus: a resource for chemical disease relation extraction. *Database*, 2016, 05 2016. baw068.

[15] Jose L. Martinez-Rodriguez, Ivan Lopez-Arevalo, and Ana B. Rios-Alvarado. Mining information from sentences through semantic web data and information extraction tasks. *Journal of Information Science*, 0(0):0165551520934387, 0.

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

[17] Michael Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov,

and Max Welling. Modeling relational data with graph convolutional networks. In Aldo Gangemi, Roberto Navigli, Maria-Esther Vidal, Pascal Hitzler, Raphaël Troncy, Laura Hollink, Anna Tordai, and Mehwish Alam, editors, *The Semantic Web*, pages 593–607, Cham, 2018. Springer International Publishing.

[18] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[19] Ye Wu, Ruibang Luo, Henry C. M. Leung, Hing-Fung Ting, and Tak-Wah Lam. Renet: A deep learning approach for extracting gene-disease associations from literature. In Lenore J. Cowen, editor, *Research in Computational Molecular Biology*, pages 272–284, Cham, 2019. Springer International Publishing.

[20] Chaojun Xiao, Yuan Yao, Ruobing Xie, Xu Han, Zhiyuan Liu, Maosong Sun, Fen Lin, and Leyu Lin. Denoising relation extraction from document-level distant supervision. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3683–3688, Online, nov 2020. Association for Computational Linguistics.

[21] Benfeng Xu, Quan Wang, Yajuan Lyu, Yong Zhu, and Zhendong Mao. Entity Structure Within and Throughout: Modeling Mention Dependencies for Document-Level Relation Extraction. *arXiv e-prints*, page arXiv:2102.10249, feb 2021.

[22] Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. DocRED: A large-scale document-level relation extraction dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 764–777, Florence, Italy, jul 2019. Association for Computational Linguistics.

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

[24] Zhenyu Zhang, Bowen Yu, Xiaobo Shu, Tingwen Liu, Hengzhu Tang, Wang Yubin, and Li Guo. Document-level relation extraction with dual-tier heterogeneous graph. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1630–1641, Barcelona, Spain (Online), dec 2020. International Committee on Computational Linguistics.

[25] Wenxuan Zhou, Kevin Huang, Tengyu Ma, and Jing Huang. Document-Level Relation Extraction with Adaptive Thresholding and Localized Context Pooling. *arXiv e-prints*, page arXiv:2010.11304, oct 2020.